



BSI Standards Publication

Date and time — Representations for information interchange

Part 2: Extensions

National foreword

This British Standard is the UK implementation of [ISO 8601-2:2019](#). Together with [BS ISO 8601-1:2019](#), it supersedes [BS ISO 8601:2004](#), which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee IST/47/-/3, eBusiness.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2019
Published by BSI Standards Limited 2019

ISBN 978 0 580 93759 0

ICS 01.140.30

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 28 February 2019.

Amendments/corrigenda issued since publication

Date	Text affected
------	---------------

INTERNATIONAL STANDARD

ISO
8601-2

First edition
2019-02-26

Date and time — Representations for information interchange —

Part 2: Extensions

*Date et heure — Représentations pour l'échange d'information —
Partie 2: Extensions*



Reference number
ISO 8601-2:2019(E)

© ISO 2019



COPYRIGHT PROTECTED DOCUMENT

© ISO 2019, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Page

Foreword	vii
Introduction	viii
1 Scope	1
2 Normative references	1
3 Terms, definitions, symbols and abbreviated terms	1
3.1 Terms and definitions	2
3.1.1 Basic concepts	2
3.1.2 Feature description	2
3.1.3 Seasons	4
3.2 Symbols and abbreviated terms	4
3.2.1 General	4
3.2.2 Time scale component symbols	4
3.2.3 Composite component symbols	5
3.2.4 Symbols used to represent time scale component features	6
3.2.5 Symbols used in date and time representations	6
3.2.6 Designator symbols used in date and time expressions	6
3.2.7 Component symbols, representations and expressions	7
4 Extensions to time scale components and units	8
4.1 General	8
4.2 Order of time scale units	8
4.3 Additional explicit forms	8
4.3.1 General	8
4.3.2 Value prefixing	8
4.3.3 Calendar day of week	8
4.3.4 Calendar day of year	9
4.3.5 Decade	9
4.3.6 Century	9
4.4 Numerical extensions	9
4.4.1 Negative values	9
4.4.2 Exponential values	12
4.4.3 Significant digits	13
4.5 Qualification of uncertainty and approximation	13
4.6 Unspecified digits	14
4.6.1 General	14
4.6.2 Unspecified time component value in explicit forms	14
4.6.3 Unspecified time component digits in implicit forms	14
4.7 Expanded calendar year	15
4.7.1 General	15
4.7.2 Letter-prefixed calendar year	15
4.7.3 Exponential calendar year	15
4.7.4 Significant digits	15
4.8 Sub-year groupings	16
4.8.1 Listing of seasons and common sub-year groupings	16
4.8.2 Groupings represented as time scale components	16
4.8.3 Groupings represented as months	17
5 Grouped time scale units	17
5.1 General	17
5.2 Unit definition	17
5.3 Unit value	18
5.4 Application within representations	18
5.4.1 General	18
5.4.2 Use of grouped units	19

5.4.3	Adherence to grouped unit boundaries.....	20
5.4.4	Representation with time shift.....	20
5.4.5	Conversion to basic time scale units.....	20
6	Set representation.....	21
6.1	Set of date and time expressions.....	21
6.2	Single element amongst set.....	21
6.3	Range element expansion.....	21
6.4	Set representations and expansion.....	22
6.5	Expressions with time scale components.....	22
6.6	Integer expressions.....	22
7	Explicit representation for date and time.....	23
7.1	General.....	23
7.2	Date 23.....	
7.2.1	General.....	23
7.2.2	Calendar date.....	23
7.2.3	Ordinal date.....	23
7.2.4	Week date.....	23
7.3	Time of day.....	23
7.3.1	Local time of day.....	23
7.3.2	Beginning of the day.....	24
7.4	Time shift.....	24
7.5	Date with shift.....	25
7.6	Time of day with time shift.....	25
7.7	Date and time of day.....	25
7.7.1	General.....	25
7.7.2	Date and time only.....	25
7.7.3	Date and time with shift.....	25
7.8	Decade.....	25
7.9	Century.....	26
7.10	Omission of zero-valued components.....	26
7.11	Indication of precision.....	26
7.12	Decimal fractions for time.....	26
7.13	Representations other than complete.....	26
7.14	Time intervals.....	27
7.14.1	General.....	27
7.14.2	Time scale component order.....	27
7.14.3	Time shift indication.....	27
7.15	Recurring time intervals.....	27
8	Qualification of date and time expressions.....	28
8.1	General.....	28
8.2	Principles.....	28
8.2.1	Complete qualification.....	28
8.2.2	Group qualification.....	28
8.2.3	Individual qualification.....	28
8.2.4	Preferred representations for resolving ambiguity.....	28
8.3	Time scale components allowing qualification.....	29
8.3.1	General.....	29
8.3.2	Calendar year, left qualified:.....	29
8.3.3	Calendar month, left qualified.....	29
8.3.4	Calendar week of year, left qualified.....	29
8.3.5	Calendar day of month, left qualified.....	29
8.3.6	Calendar day of week, left qualified.....	30
8.3.7	Calendar day of year, left qualified.....	30
8.3.8	Clock hour, left qualified.....	30
8.3.9	Clock minute, left qualified.....	30
8.3.10	Clock second, left qualified.....	31
8.3.11	Decade, left qualified.....	31

8.3.12	Century, left qualified.....	31
8.4	Calendar date representations with qualification.....	31
8.4.1	Complete representation of a calendar date.....	31
8.4.2	Representations of calendar dates with reduced precision.....	32
8.4.3	Expanded representations of calendar dates.....	33
8.4.4	Qualification of a group of time scale components from the right.....	34
8.4.5	Qualification of individual time scale components.....	34
8.4.6	Allowing group and individual qualifications of time scale components.....	35
8.5	Date and time expressions with qualification.....	35
9	Unspecified digits.....	35
9.1	General.....	35
9.2	Calendar date representations with unspecified digits.....	36
9.2.1	Unspecified time scale component values from the right.....	36
9.2.2	Unspecified digit anywhere in time scale component.....	38
9.3	Date and time expressions with unspecified digits.....	38
10	Extended time interval representations.....	39
10.1	General.....	39
10.2	Unknown or open start or end time intervals.....	39
10.3	Qualification of dates in time intervals.....	40
10.3.1	General.....	40
10.3.2	Complete qualification.....	40
10.3.3	Partial qualification.....	40
10.4	Unspecified portions of dates in time intervals.....	40
10.5	Uncertain and approximate dates in unknown or open time intervals.....	40
10.6	Before and after with qualified time scale components.....	40
11	Explicit duration and extensions.....	41
11.1	General.....	41
11.2	Durational units.....	41
11.3	Representations.....	41
11.3.1	General.....	41
11.3.2	Composite representation.....	41
11.3.3	Precedence representation.....	42
11.4	Fractional duration.....	43
12	Selection of date and time.....	43
12.1	General.....	43
12.2	Selection rules.....	44
12.2.1	Selection of calendar month of year.....	44
12.2.2	Selection of calendar week of year.....	44
12.3	Selection of calendar day of month.....	45
12.4	Selection of week days.....	45
12.5	Selection of ordinal days in calendar year.....	45
12.6	Selection of hours.....	45
12.7	Selection of minutes.....	46
12.8	Selection of seconds.....	46
12.9	Selection of position.....	46
12.10	Selection with time interval.....	47
12.11	Application within representations.....	47
12.11.1	General.....	47
12.11.2	Context set by selection.....	48
12.11.3	Within time intervals.....	48
13	Recurring time intervals with repeat rules.....	48
13.1	General.....	48
13.2	Method of specification.....	49
13.3	Specification of time interval.....	49
13.4	Repeat rule.....	49
13.4.1	General.....	49

13.4.2	Eligibility part and eligible time intervals	49
13.4.3	Selection part and selection rules	50
13.5	Representations	50
13.6	Evaluation of a repeat rule	51
13.6.1	General	51
13.6.2	Time scale unit precision	52
13.6.3	Inheritance of component values from time interval start	52
14	Date and time arithmetic	53
14.1	General	53
14.2	Addition and subtraction	53
14.3	Multiplication	54
14.4	Date time modified by duration	55
15	Profiles	56
15.1	General	56
15.2	Requirements	56
Annex A (informative) Profile: Extended Date/Time Format		57
Annex B (informative) Interactions between eligible time intervals with the selection part		63
Annex C (informative) Compatibility considerations of repeat rules with IETF RFC 5545 recurrences		66
Annex D (informative) Evaluation of date time formulas and duration considerations		68
Bibliography		75

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 154, *Processes, data elements and documents in commerce, industry and administration*.

This first edition of [ISO 8601-2](#), together with [ISO 8601-1](#), cancels and replaces [ISO 8601:2004](#), which has been technically revised.

The main changes compared to [ISO 8601:2004](#) are as follows:

- addition of standard representations for concepts not previously supported, such as negative values, qualification, sets, seasons, extended time intervals, selections, grouped units, repeating rules and profiles.

A list of all parts in the [ISO 8601](#) series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

The purpose of this document is to provide a set of date and time format representations for information interchange beyond those supported by [ISO 8601-1](#).

There are various concepts and representations that many applications find useful not supported by [ISO 8601-1](#), including:

- negative values for time scale components;
- qualification of expressions and time scale components;
- set representation of date and time expressions;
- seasons;
- time intervals with open or unknown starts or ends;
- expression of movable days through date and time selection;
- date and time expressions without digit length limits; and
- profiles for specifying feature support amongst [ISO 8601](#) (all parts) features.

Such concepts are often represented according to various ad-hoc conventions; this document aims to provide a standard syntax for their representation.

The extended representations allow unambiguous interpretation, enforce the confidence of interoperability and minimize the risk of misinterpretations and their consequences.

Date and time — Representations for information interchange —

Part 2: Extensions

1 Scope

This document specifies additional representations of dates of the Gregorian calendar and times based on the 24-hour clock that extend the basic rules and composite elements of those defined in [ISO 8601-1](#). These representations are specified as character strings for use in information interchange. It is also applicable for representing times and time shifts based on Coordinated Universal Time (UTC).

These extensions include:

- uncertain or approximate dates, or dates with portions unspecified;
- extended time intervals;
- divisions of a year;
- sets and choices of calendar dates;
- grouped time scale units;
- repeat rules for recurring time intervals; and
- date and time arithmetic.

This document excludes the representation of date elements from non-Gregorian calendars, or times not from the 24-hour clock. This document does not address character encoding of representations specified in this document.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

[ISO 8601-1:2019](#), *Date and time — Representation for information interchange — Part 1: Basic rules*

3 Terms, definitions, symbols and abbreviated terms

For the purposes of this document, the terms and definitions given in [ISO 8601-1](#) and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <http://www.electropedia.org/>

3.1 Terms and definitions

3.1.1 Basic concepts

3.1.1.1

approximate date

calendar date which is an estimate whose value is asserted to be possibly correct

Note 1 to entry: The degree of confidence in approximation depends on the application.

3.1.1.2

uncertain date

calendar date whose source is considered *dubious* ([3.1.1.4](#))

3.1.1.3

date with unspecified part

calendar date of which a part is unstated

Note 1 to entry: The unstated part can be year, year and month, month, month and day, or year and day. It is unstated because it has not (yet) been assigned (it can be assigned in the future), or because it is classified or unknown, or for any other reason.

3.1.1.4

dubious

not to be relied upon

3.1.1.5

qualification symbol

symbol that indicates certain qualification to a value it applies to

EXAMPLE The symbol [?] indicates that the value it applies to is uncertain.

3.1.1.6

movable day

repeatedly occurring day in a calendar that is represented by criteria set by one or more *selection rules* ([3.1.2.6](#)), but does not always resolve to a fixed calendar day in every repeating cycle

EXAMPLE 1 Thanksgiving Day in the U.S. and Canada are considered movable days, set to the fourth Thursday in November and the second Monday in October, respectively.

EXAMPLE 2 First Day of Summer (*sumardagurinn fyrsti*) in Iceland is considered a movable day, set to the first Thursday after April 18th.

EXAMPLE 3 Father's Day is celebrated on the third Sunday in June in North America and is considered a movable day.

Note 1 to entry: The selection of this term pays homage to the phrase "movable feast" used in some calendars to indicate the rule-based determination of annual events.

3.1.1.7

negative duration

duration in the reverse direction to the proceeding time scale

3.1.2 Feature description

3.1.2.1

feature

single function or group of functions

3.1.2.2

conformity level

value assigned to a subset of *features* ([3.1.2.1](#)) within a *profile* ([3.1.2.3](#))

Note 1 to entry: A profile may refer to these conformity levels to facilitate the specification of conformance to the profile.

3.1.2.3

profile

subset of *features* ([3.1.2.1](#)) described in a standard or a set of standards

Note 1 to entry: A community may develop a profile to describe how to carry out functions or apply features specified in a standard or family of standards in a manner relevant to that community.

Note 2 to entry: A profile often describes what features are to be supported and how to apply those features. In cases where multiple methods are allowed for a required feature, a profile may select a single method. In cases where a particular function allows different interpretations, a profile may select a single interpretation or provide clarification. A profile may list out features that need not be supported. It may specify several *conformity levels* ([3.1.2.2](#)).

Note 3 to entry: The term “profile” is used in this document with the specific definition (including the notes) given here, despite its various meanings in other contexts.

3.1.2.4

ISO 8601 profile

profile ([3.1.2.3](#)) whose base standard is the [ISO 8601](#) series

3.1.2.5

repeat rule

set of *eligible time intervals* ([3.1.2.8](#)) and *selection rules* ([3.1.2.6](#)) that allows computation of a set of *matching instants* ([3.1.2.9](#))

3.1.2.6

selection rule

rule specifying restrictions on the value of a time scale component

3.1.2.7

repeat cycle

set of repeating instants, calculated by a specified start instant and specified duration gap between the repeating instants

3.1.2.8

eligible time interval

time interval eligible for matching using *selection rules* ([3.1.2.6](#))

3.1.2.9

matching instants

set of instants, computed by a *repeat rule* ([3.1.2.5](#)), that belongs within *eligible time intervals* ([3.1.2.8](#)) and fulfils criteria set by specified *selection rules* ([3.1.2.6](#))

3.1.2.10

date time formula

expression that specifies modification of a date and time expression with a duration

3.1.2.11

durational unit

time scale component that composes the duration date and time representation

3.1.2.12

grouped time scale unit

time scale unit composed of a duration expressed by one or more *durational units* ([3.1.2.11](#))

3.1.2.13

equation of time

difference between mean solar time and apparent solar time, which varies with time within a calendar year

Note 1 to entry: A clock is a type of device that measures mean solar time; a sundial is a type of device that measures apparent solar time.

3.1.3 Seasons

3.1.3.1

season

time interval resulting from the common division of a calendar year into four time intervals, *spring* ([3.1.3.2](#)), *summer* ([3.1.3.3](#)), *autumn* ([3.1.3.4](#)) or *winter* ([3.1.3.5](#))

Note 1 to entry: A single calendar date may represent different seasons depending on local customs or location, such as the difference between the northern or southern hemispheres.

3.1.3.2

spring

season ([3.1.3.1](#)) following *winter* ([3.1.3.5](#)) and preceding *summer* ([3.1.3.3](#))

3.1.3.3

summer

season ([3.1.3.1](#)) following *spring* ([3.1.3.2](#)) and preceding *autumn* ([3.1.3.4](#))

3.1.3.4

autumn

season ([3.1.3.1](#)) following *summer* ([3.1.3.3](#)) and preceding *winter* ([3.1.3.5](#))

3.1.3.5

winter

season ([3.1.3.1](#)) following *autumn* ([3.1.3.4](#)) and preceding *spring* ([3.1.3.2](#))

3.2 Symbols and abbreviated terms

3.2.1 General

Representations and expressions specified in this document make use of the symbols given in [ISO 8601-1](#) and the following.

3.2.2 Time scale component symbols

c(feat)	the left side of a representation statement where "c" is a time scale component (e.g. "year") and "feat" is a feature, for example "q" or "m" (see below)
yearE	time scale component calendar year in explicit form, specified in ISO 8601-1:2019 , 4.3.2 b)
monthE	time scale component calendar month in explicit form, specified in ISO 8601-1:2019 , 4.3.3 b)
weekE	time scale component calendar week of year in explicit form, specified in ISO 8601-1:2019 , 4.3.4 b)
dayE	time scale component calendar day of month in explicit form, specified in ISO 8601-1:2019 , 4.3.5 b)
daykE	time scale component calendar day of week in explicit form, specified in 4.3.3

dayoE	time scale component calendar day of year in explicit form, specified in 4.3.4
hourE	time scale component hour in explicit form, specified in ISO 8601-1:2019 , 4.3.8 b)
minE	time scale component minute in explicit form, specified in ISO 8601-1:2019 , 4.3.9 b)
secE	time scale component second in explicit form, specified in ISO 8601-1:2019 , 4.3.10 b)
decE	time scale component decade (defined in ISO 8601-1:2019 , 3.1.2.22) in explicit form, specified in 4.3.5
centE	time scale component century (defined in ISO 8601-1:2019 , 3.1.2.23) in explicit form, specified in 4.3.6
season	the values 21 to 24 as specified in 4.8.1
ydivision	the values 25 to 41 as specified in 4.8.1
seasonE	the values 21 to 41 as specified in 4.8.1
expr <i>i</i>	any date and time expression, where <i>i</i> is empty or a positive integer
intexpr	a date and time expression accepted by the time interval representation as start or end as determined in ISO 8601-1:2019 , 5.4.3.

3.2.3 Composite component symbols

dateI	date representation in implicit form, specified in ISO 8601-1:2019 , 3.2.3 as date and dateX
timeI	time of day representation in implicit form, specified in ISO 8601-1:2019 , 3.2.3 as time and timeX
timeE	time of day representation in explicit form
dtE	date and time representation in explicit form
dsE	date with shift representation in explicit form
dtsE	date and time with shift representation in explicit form
tiseE	time interval representation with a start and end in explicit form
tisdE	time interval representation with a start in explicit form and duration
tiedE	time interval representation with a duration and end in explicit form
positiveDuration	representation of [duration] specified in ISO 8601-1:2019 , 5.4.2 that contains only time scale components that have positive values
timeUnits	time scale components for time of day representation
groupUnit	grouped time scale unit representation
group	grouped time scale unit representation with value
groupDateTime	date time representation that incorporates grouped time scale units
timeIntervalE	collective representation of tiseE, tisdE, and tiedE

3.2.4 Symbols used to represent time scale component features

q	qualified form for time scale components, as specified in 4.5
p	prefixed year, only used in accordance with requirements specified in 4.7.2
e	exponential form for time scale components, as specified in 4.7.3
s	significant digits form for time scale components, as specified in 4.4.3
m	minus form to allow time scale components to accept negative integer values, as specified in 4.4.1

3.2.5 Symbols used in date and time representations

!	symbol indicating that the token following this symbol is optional (may be omitted)
Q	placeholder for a qualification modifier which applies to the time scale component value
AA	the grouping digit pair, representing a two-digit integer that is 21 or greater
I	a positive integer
negi	a positive or negative integer; equivalent to [!]["-"][i]

3.2.6 Designator symbols used in date and time expressions

"-"	a minus sign represented by the character "-" to indicate a negative value
"K"	the calendar day of week designator, represented by the character "K", preceding a data element which represents the ordinal number of a calendar day within a calendar week
"J"	the decade designator, represented by the character "J", preceding a data element which represents the decade number
"C"	the century designator, represented by the character "C", preceding a data element which represents the century number
"O"	the calendar day of year designator, represented by the character "O", preceding a data element which represents the ordinal number of a calendar day within a calendar year
"B"	the suffix designator to represent years before year one (1), represented by the character "B", placed after the time scale components of calendar year, decade and century
"E"	the exponent designator, represented by the character "E", preceding the component which represents the exponential part of a time scale component value
"S"	the significant digit designator, represented by the character "S", preceding the component which represents the number of significant digits of the time scale component value
"G"	the grouped time scale unit prefix designator, represented by the character "G", indicating that a grouping calculation applies to the subsequent time scale components until encountering the corresponding suffix designator
"U"	the grouped time scale unit suffix designator, represented by the character "U", indicating that a grouping calculation applies to the preceding time scale components from the corresponding prefix designator

"I"	the instance designator, represented by the character "I", indicating that a specific instance is to be selected within the time scale component
"F"	the frequency designator, represented by the character "F", preceding the component which represents the frequency part of a repeat rule
"L"	the selection prefix designator, represented by the character "L", preceding the component which represents the selection part of a repeat rule until encountering the corresponding suffix designator
"N"	the selection suffix designator, represented by the character "N", following the component which represents the selection part of a repeat rule from the corresponding prefix designator
"X"	the unspecified digit, used within a date with unspecified part, represented by the character "X", indicating that the time scale component value of the specific digit it replaces is unspecified
"*"	the unspecified digit modifier, represented by the character "*", indicating, when used with the unspecified digit as "X*", that the entire time scale component value it applies to is unspecified
"?"	qualification modifier, represented by the character "?", indicating that the time scale component value it applies to is uncertain
"~"	qualification modifier, represented by the character "~", indicating that the time scale component value it applies to is approximate
"%"	qualification modifier, represented by the character "%", indicating that the time scale component value it applies to is both uncertain and approximate
"..x"	indicating "on or before" value x if applied as a prefix to a time scale component
x".."	indicating "on or after" value x if applied as a suffix to a time scale component
x"..y"	indicating "between x and y" (inclusive) when applied between two time scale components

NOTE 1 The designator symbol "J" is assigned from its ordinal position being the tenth letter of the English alphabet.

NOTE 2 The designator symbols "G" and "U" are used to demarcate the "grouped time scale unit" time scale components as they are the initial letters of the words "group" and "unit".

3.2.7 Component symbols, representations and expressions

Throughout this document are statements of the form:

$$\langle \text{component symbol} \rangle = \langle \text{representation} \rangle$$

This means that the <component symbol> is assigned a representation <representation>, where the <representation> describes accepted symbols.

EXAMPLE The statement 'year(q) = [!][Q][YYYY]' means that the component symbol "year(q)" is represented as "[!][Q][YYYY]", year(q) is the symbol for "qualified form for year", and "[!][Q][YYYY]" means the string "YYYY" optionally preceded by a qualification character. Thus, for year 1985 and qualification character '?', the representation [!][Q][YYYY], would result in the expression '1985' or '?1985'.

4 Extensions to time scale components and units

4.1 General

Units of time scales specified in [ISO 8601-1](#) are extended according to rules specified in this clause.

Features described in this clause that apply to implicit forms (e.g., [dateI]) shall only be used together with the extended formats (where separators are placed between time scale components); these features shall not be used with the basic formats of the implicit forms.

4.2 Order of time scale units

This document refers to an "order" amongst time scale units, which is defined by the duration of a time scale unit relative to other time scale units. For example, since a single calendar month has a duration always greater than a single unit of calendar day, the former is considered a higher order time scale unit than the latter.

The relative order of commonly used time scale units is specified below, with the higher listed entry considered a higher order time scale unit than a lower listed entry:

- calendar century;
- calendar decade;
- calendar year;
- calendar month;
- calendar week;
- calendar day;
- clock hour;
- clock minute;
- clock second.

NOTE a higher order time scale does not necessarily mean its duration can be expressed as an integer multiple of a lower order time scale. For example, a calendar month can have 28, 29, 30 or 31 days and cannot always be expressed as an integer multiple of a calendar week (7 days).

4.3 Additional explicit forms

4.3.1 General

In [ISO 8601-1](#), explicit forms of time scale components are used for the representation of duration only. This subclause extends usage of explicit forms for the representation of date and time.

4.3.2 Value prefixing

Prefixing of the value with one or more "0"s (the number zero) to the integer [i] is allowed in the explicit forms of time scale components.

EXAMPLE '1M', '01M' and '0001M' all describe the calendar month January.

4.3.3 Calendar day of week

The calendar day of week is represented below, as an extension to [ISO 8601-1:2019](#), 4.3.6:

Explicit:

[i]["K"]

EXAMPLE '1K' in explicit form expresses the week day Monday.

4.3.4 Calendar day of year

The calendar day of year is represented below, as an extension to [ISO 8601-1:2019](#), 4.3.7:

Explicit:

[i]["O"]

EXAMPLE '350O' in explicit form expresses the ordinal day 350 of a year.

4.3.5 Decade

The decade is represented below as an extension to [ISO 8601-1:2019](#), 4.3.11:

Explicit:

[i]["J"]

EXAMPLE 1 '196J' in explicit form expresses the decade of the 1960s, spanning the calendar years 1960 to 1969.

EXAMPLE 2 '0J' in explicit form expresses the decade spanning calendar years 0 to 9.

4.3.6 Century

The century is represented below as an extension to [ISO 8601-1:2019](#), 4.3.12:

Explicit:

[i]["C"]

EXAMPLE 1 '16C' in explicit form expresses the century of the 1600s, spanning the calendar years 1600 to 1699.

EXAMPLE 2 '0C' in explicit form expresses the century spanning the calendar years 00 to 99.

4.4 Numerical extensions

4.4.1 Negative values

4.4.1.1 General

Certain time scale components are allowed to accept a negative integer as their value. The following symbol is used in representations that accept a positive or negative integer.

negi = [!]["-"][i]

In order to convert a time scale component to accept negative values, the following representation is used.

c(m) = [negi][c]

where

[c] is the time scale component representation that accepts non-negative values only;

[m] is a feature label extending the time scale component [c] to accept negative values.

This representation only applies to time scale components that describe a time scale unit.

EXAMPLE If "c" is the time scale component [yearE] in explicit form (see [ISO 8601-1:2019](#), 4.3.2, b)), c(m) means "calendar year accepting a negative value". The explicit year representation [i]["Y"] becomes [negi]["Y"] when [i] is replaced with [negi], and expands to [!]["-"][i]["Y"], which accepts the representations [i]["Y"] or ["-"][i]["Y"]. In expressions of this representation, the value of [i] or ["-"][i] represents a non-zero integer whose value identifies a calendar year.

4.4.1.2 Calendar year

In [ISO 8601-1](#), a negative calendar year cannot be expressed for the years -0001 through -9999. This subclause removes that restriction.

The calendar year is represented as follows, where a) and b) extend [ISO 8601-1:2019](#), 4.3.2, allowing negative values for years prior to year zero (0):

a) Implicit:

$\text{year}(m) = [!]["-"][\text{year}]$

EXAMPLE 1 '-1985' in implicit form expresses the calendar year -1985.

b) Explicit:

$\text{yearE}(m) = [!]["-"][\text{yearE}]$

EXAMPLE 2 '-2018Y' in explicit form expresses the calendar year -2018.

When a negative calendar year represents a date, the negative value is to represent the number of years prior to year zero (0) (year zero is expressed as '0000' in implicit form and '0Y' in explicit form).

EXAMPLE 3 The year immediately preceding year zero (0) is expressed as '-0001' in implicit form and '-1Y' in explicit form.

4.4.1.3 Calendar week of year

The calendar week of year is represented as follows extending [ISO 8601-1:2019](#), 4.3.4:

Explicit:

$\text{weekE}(m) = [!]["-"][\text{weekE}]$

When a negative calendar week of year represents a date, the negative value expresses the number of weeks counting in reverse from the end of the calendar year, where the last week of the year is represented by the integer value '-1'.

EXAMPLE '-3W' expresses the third last week of the year.

4.4.1.4 Calendar day of month

The calendar day of month is represented as follows extending [ISO 8601-1:2019](#), 4.3.5:

Explicit:

$\text{dayE}(m) = [!]["-"][\text{dayE}]$

When a negative calendar day of month represents a date, the negative value expresses the number of days counting in reverse from the end of the calendar month, where the last day of the month is represented by the integer value '-1'.

EXAMPLE '-5D' represents the fifth last day of the month.

4.4.1.5 Calendar day of year

The calendar day of year is represented as follows extending [4.3.4](#):

Explicit:

$$\text{dayoE}(m) = [!][\text{"-"}][\text{dayoE}]$$

When a negative calendar day of year represents a date, the negative value expresses the number of days counting in reverse from the end of the calendar year, where the last day of the year is represented by the integer value '-1'.

EXAMPLE 1 '-70' represents the seventh last day of the calendar year, which is always December 25th.

EXAMPLE 2 '-3060' represents February 29th for a leap year and February 28th otherwise.

4.4.1.6 Calendar year before year one (1)

The calendar year before year one (1) is represented as follows:

Explicit:

$$[\text{yearE}][\text{"B"}]$$

EXAMPLE 1 '1YB' the first year before year one (1), equivalent to the effect of '0Y'.

EXAMPLE 2 '12YB' the twelfth year before year one (1), equivalent to the effect of '-11Y'.

4.4.1.7 Calendar decade before year one (1)

The calendar decade before year one (1) extends the calendar decade's implicit form (specified in [ISO 8601-1:2019](#), 4.3.11) and its explicit form (specified in [4.3.5](#)) to accept negative values. It identifies a decade consisting of all ten years identified by the provided digits with an additional digit appended.

In implicit form, the calendar decade before year one (1) is specified by a three-digit negative integer. There is no digit limitation in explicit form.

The respective forms are represented as follows:

a) Implicit:

$$\text{dec}(m) = [!][\text{"-"}][\text{dec}]$$

EXAMPLE 1 '-019' in implicit form expresses the years -199 through -190.

b) Explicit:

$$\text{decE}(m) = [!][\text{"-"}][\text{decE}]$$

EXAMPLE 2 '-12J' in explicit form expresses the years -129 through -120.

EXAMPLE 3 The "negative zero decade" is identified by '-000' in implicit form ('-0J' in explicit form), which expresses year -9 through year zero (0).

NOTE Zero decade and negative zero decade are not the same decade. They overlap; both include year zero (0).

4.4.1.8 Calendar century before year one (1)

The calendar century before year one (1) extends the calendar century's implicit form (specified in [ISO 8601-1:2019](#), 4.3.2) and its explicit form (specified in [4.3.6](#)) to accept negative values. It identifies a century consisting of all 100 years identified by those digits with an additional two digits appended.

In implicit form, the calendar decade before year one (1) is specified by a two-digit negative integer. There is no digit limitation in explicit form.

The respective forms are represented as follows:

a) Implicit:

$\text{cent}(m) = [!]["-"][\text{cent}]$

EXAMPLE 1 '-19' in implicit form expresses the years -1999 through -1900.

b) Explicit:

$\text{centE}(m) = [!]["-"][\text{centE}]$

EXAMPLE 2 '-12C' in explicit form expresses the years -1299 through -1200.

EXAMPLE 3 The "negative zero century" is identified by '-00' in implicit form ('-0C' in explicit form), which expresses year -99 through year zero (0).

NOTE Zero century and negative zero century are not the same century. They overlap; both include year zero (0).

4.4.1.9 Duration

A duration in the reverse direction, called a "negative duration" in this document, can be expressed using the following representation based on the [duration] representation specified in [ISO 8601-1:2019](#), 5.5.2. In this case, all time scale components within the duration representation shall be positive.

$\text{duration}(m) = [!]["-"][\text{positiveDuration}]$

where [positiveDuration] is the representation of a positive duration.

EXAMPLE 1 '-P100D' in date represents the duration of 100 days in the reverse direction. The duration formula 'P365D - P100D' results in 'P265D'.

EXAMPLE 2 '-P1Y3D' in date represents the duration of one year and three days in the reverse direction. The duration formula 'P5Y6D - P1Y3D' results in 'P4Y3D'.

4.4.2 Exponential values

In some cases, the representation of a value scale is more important than the need for accuracy. The following representation is used for a value that accepts an exponential value. This mechanism originates from Library of Congress EDTF.

$\text{exp} = [\text{significand}]["E"][\text{exponent}]$

where

[significand] is an integer value of representation [negi];

[exponent] is an integer value of representation [i];

[exp] provides the resulting value of raising the [significand] (a positive or negative integer) to the [exponent] (a positive integer used as a power of 10).

EXAMPLE 1 '3E3' (the value $3 \times 10^3 = 3\,000$).

EXAMPLE 2 '-82E9' (the value $-82 \times 10^9 = -82\,000\,000\,000$).

A time scale component that accepts unbounded integers (such as calendar year), or is used to describe duration (such as calendar month), can be converted to accept an exponential value using the following representation, where c denotes a time scale component.

$c(e) = [c]$, where the representation of $[i]$ is replaced with $[exp]$

where

$[c]$ indicates the time scale component representation;

$[e]$ indicates that the time scale component accepts exponential values.

EXAMPLE 3 If "c" is the time scale component yearE in explicit form, $c(e)$ means "calendar year accepting an exponential value". The explicit year representation $[i][Y]$ becomes $[exp][Y]$ when $[i]$ is replaced with $[exp]$, and expands to $[significand][E][exponent][Y]$ and finally $[negi][E][i][Y]$. In this representation, the value of $[exp]$ represents a calendar year. '3E8Y' is an expression accepted by this representation.

4.4.3 Significant digits

When a value assigned to a time scale component is large enough, it is often useful to compact the value expression by expressing the precision of the value using the representation as follows. This mechanism originates from Library of Congress EDTF.

$sig = [value][S][precision]$

where

$[value]$ is an integer value that describes the estimated value, represented as $[negi]$;

$[precision]$ is the number of significant digits counting from the leftmost end of $[value]$, describing the precision of the value, represented as $[i]$;

$[sig]$ provides a representation of an estimated value within a value range of specified precision.

EXAMPLE 1 '1230S2', a value between 1200 and 1299, estimated to be 1230.

EXAMPLE 2 '3141592653S4', a value between 3141000000 and 3141999999, estimated to be 3141592653.

EXAMPLE 3 '271828E5S6', a value between 27182800000 and 27182899999, estimated to be 27182800000.

A time scale component that accepts unbounded integers (such as calendar year), or is used to describe duration (such as calendar month), is allowed to accept a value with specified significant digits using the following representation.

$c(s) = [c]$, where the representation of $[i]$ is replaced with $[sig]$

where

$[c]$ indicates the time scale component representation;

$[s]$ indicates that the time scale component allows significant digits.

EXAMPLE 4 If "c" is the time scale component yearE in explicit form, $c(s)$ means "calendar year accepting a value with significant digits". The explicit year representation $[i][Y]$ becomes $[sig][Y]$ when $[i]$ is replaced with $[sig]$, and expands to $[value][S][precision][Y]$ and finally $[negi][S][i][Y]$. In this expression, $[value]$ describes the estimated calendar year, $[precision]$ describes the number of digits with precision, and $[sig]$ describes the specified value range. '6633110S3Y' is an expression accepted by this representation.

4.5 Qualification of uncertainty and approximation

This subclause extends time scale components to accept qualification of uncertainty and approximation of their values.

The implicit forms (the "implied form" and "extended form" as described in [ISO 8601-1:2019](#), 4.3) and explicit form (described in [4.3](#) and [ISO 8601-1:2019](#), 4.3) of time scale components are extended with a qualifying symbol, namely, ["?"], ["~"] or ["%"] (see [3.2.6](#)), according to the following representations:

Implicit:

$$c(q) = [!][Q][c]$$

Explicit:

$$c(q) = [i][!][Q][symbol(c)]$$

where $[symbol(c)]$ is the designated symbol for the time scale component.

EXAMPLE If c is the timescale component "year", $[symbol(c)]$ is the symbol for the year time scale component — the designated symbol ["Y"]. If the qualification character is ["?"] and $[i]$ is the value '12', then the representation $[i][Q][!][symbol(c)]$ would accept the expressions '12?Y' or '12Y'.

4.6 Unspecified digits

4.6.1 General

The unspecified value character "X" is used to indicate unspecified digits or values in time scale components.

4.6.2 Unspecified time component value in explicit forms

In explicit forms of time scale components,

- the unspecified value character followed by a "*" ("X*") indicates that the entire value is unspecified;
- the unspecified value character "X" is used as a replacement for any digit in a time scale component value to indicate that the digit is unspecified.

EXAMPLE 1 '195XY' expresses an unspecified calendar year in the 1950s.

EXAMPLE 2 '1390YXXM' expresses a two-digit calendar month in 1390.

EXAMPLE 3 '2052Y1MX*D' expresses some calendar day in January 2052.

EXAMPLE 4 'XXXYX*MXD' expresses a one-digit calendar day of a calendar month in a three-digit calendar year.

EXAMPLE 5 'X*Y12M28D' expresses December 28th of an unspecified calendar year.

4.6.3 Unspecified time component digits in implicit forms

In implicit forms of time scale components, the unspecified value character can be used as a replacement for any digit in a time scale component to indicate that the digit is unspecified. To indicate that the entire component is unspecified, the unspecified value character replaces every digit of that component.

EXAMPLE 1 '199X' expresses an unspecified calendar year in the 1990s.

EXAMPLE 2 '1999-XX' expresses some calendar month in 1999.

EXAMPLE 3 '1999-01-XX' expresses some calendar day in January 1999.

EXAMPLE 4 'XXXX-01-01' expresses January 1st of an unspecified four-digit calendar year.

4.7 Expanded calendar year

4.7.1 General

[ISO 8601-1:2019](#), 5.2.2.3 allows the value of the calendar year time scale component to exceed four digits (a year after 9999 or before -9999). However, this is allowed only by mutual agreement of the communicating parties. This subclause addresses this specific limitation.

4.7.2 Letter-prefixed calendar year

An alternative time scale component for calendar year, not requiring mutual consent, is provided here as a substitute for the [year] time scale component defined in [ISO 8601-1:2019](#), 3.2.2. It should be used only for dates that include the calendar year only, and only for years later than 9999 or earlier than -9999 (otherwise the representation provided in [ISO 8601-1](#) should be used).

The character "Y" can be used to prefix a string to signify that the element is a calendar year. This time scale component is denoted as [year(p)].

$\text{year}(p) = ["Y"][\text{negi}]$

EXAMPLE 1 'Y1700000002' represents a positive calendar year 1700000002.

EXAMPLE 2 'Y-1700000002' represents a negative calendar year -1700000002.

4.7.3 Exponential calendar year

The method of [4.4.2](#) can be used for representing a year whose value scale is more important than the need for accuracy in conjunction with [4.7.2](#). This time scale component is denoted as [year(e)].

$\text{year}(e) = ["Y"][\text{negi}]["E"][i]$

EXAMPLE 1 'Y17E7' (the calendar year $17 \times 10^7 = 1700000000$).

EXAMPLE 2 'Y-17E7' (the calendar year $-17 \times 10^7 = -1700000000$).

4.7.4 Significant digits

The method of [4.4.3](#) can be applied to the year time scale component in conjunction with [4.7.2](#) as well as [4.7.3](#). This time scale component is denoted as [year(s)].

The following are acceptable methods to express a calendar year in significant digit representation:

- a) Four-digit year that accepts significant digits:

[year(s)]

EXAMPLE 1 '1950S2', some year between 1900 and 1999, estimated to be 1950.

- b) Year longer than four digits that is expressed as letter-prefixed and accepts significant digits:

[year(p, s)]

EXAMPLE 2 'Y171010000S3', some year between 171010000 and 171010999, estimated to be 171010000.

- c) Year that is expressed in exponential form and accepts significant digits:

[year(p, e, s)]

EXAMPLE 3 'Y3388E2S3', some year between 338000 and 338999, estimated to be 338800.

4.8 Sub-year groupings

4.8.1 Listing of seasons and common sub-year groupings

A sub-year grouping, which is a method of grouping time scale components of a lower order than the year time scale, can be represented as the following values:

- 21 Spring (independent of location)
- 22 Summer (independent of location)
- 23 Autumn (independent of location)
- 24 Winter (independent of location)
- 25 Spring — Northern Hemisphere
- 26 Summer — Northern Hemisphere
- 27 Autumn — Northern Hemisphere
- 28 Winter — Northern Hemisphere
- 29 Spring — Southern Hemisphere
- 30 Summer — Southern Hemisphere
- 31 Autumn — Southern Hemisphere
- 32 Winter — Southern Hemisphere
- 33 Quarter 1 (3 months in duration)
- 34 Quarter 2 (3 months in duration)
- 35 Quarter 3 (3 months in duration)
- 36 Quarter 4 (3 months in duration)
- 37 Quadrimester 1 (4 months in duration)
- 38 Quadrimester 2 (4 months in duration)
- 39 Quadrimester 3 (4 months in duration)
- 40 Semestral 1 (6 months in duration)
- 41 Semestral 2 (6 months in duration)

The values '21' to '24' indicates a sub-year grouping as seasons, independent of the occurring location.

4.8.2 Groupings represented as time scale components

The following time scale components are used to represent valid ranges of the values in [4.8.1](#).

- a) Implicit, basic sub-year groupings:
season = [AA], where the range is [21] to [24]
- b) Implicit, advanced sub-year groupings:
ydivision = [AA], where the range is [21] to [41]

- c) Explicit, sub-year groupings:

seasonE = [i]["A"], where [i] is in the range from 21 to 41

4.8.3 Groupings represented as months

In a year-and-month expression (e.g. [ISO 8601-1:2019](#), 5.2.2.2), the calendar month component is extended to accept values specified in [4.8.1](#) (in addition to calendar month values) to signify sub-year groupings.

In these representations the month time scale component, [month], is replaced with [season] or [ydivision].

- a) Basic seasons in the month time scale component:

[year]["-"][season]

EXAMPLE 1 '2001-21' expresses "Spring of 2001".

- b) Hemispheric seasons and typical divisions in the month time scale component:

[year]["-"][ydivision]

EXAMPLE 2 '2001-34' expresses "second quarter of 2001".

5 Grouped time scale units

5.1 General

This clause is a generalization of sub-year groupings described in [4.8](#).

A time scale unit can be replaced by a "grouped unit" as described in this clause, based on CalConnect CC 18011, Clause 5, "Grouped time scale units".

5.2 Unit definition

A grouped time scale unit ("grouped unit") defines a time scale unit that is a composite of smaller time scale units. Its representation is provided as follows and is denoted as [groupUnit] in this document.

groupUnit = ["G"][durationUnits]["U"]

where

["G"] is the grouping prefix designator;

["U"] is the grouping suffix designator;

[durationUnits] is one or more time scale components for expressing positive duration as specified in [11.2](#).

NOTE 1 [durationUnits] is essentially a duration statement without the "P" prefix. "P" is unnecessary because the duration is always expressed between the grouping prefix and suffix, namely, "G" and "U"..

NOTE 2 When expressing the month component as a two-digit integer as described in [4.8](#), this representation does not apply, and the month component can consist entirely of two digits.

Grouped units are continuous; there is no gap between two adjacent grouped units.

EXAMPLE 1 'G11DU', every unit of the grouped unit is of eleven days duration. For example, if this grouped unit replaces the month component in a calendar date, it represents three units of the month: days 1 through 11, days 12 through 22, and days 23 through the end of month, which could be 28, 29, 30 or 31 depending on the actual month the grouping applies to.

NOTE 3 The mechanism for selecting a specific unit from the set of grouped units is specified in 5.3. Thus in this example, if group 1 is selected, it represents the first 11 day period in the month.

EXAMPLE 2 'GT30MU', every unit of the grouped unit is of thirty minutes duration.

EXAMPLE 3 'G2DT6HU', every unit of the grouped unit is of two days and six hours duration.

5.3 Unit value

The value of a grouped time scale unit is given with the following representation. This representation is denoted as [group] in this document.

group = [i][groupUnit]

where

[i] indicates the coefficient value of the grouped time scale unit;

[groupUnit] is as defined in 5.2.

EXAMPLE 1 '5G10DU', the fifth unit of ten-day groups.

EXAMPLE 2 '20GT30MU', the twentieth unit of thirty-minute groups.

EXAMPLE 3 '2G2DT6HU', the second unit of two-day groups.

The yearly division values from [33] to [41] in 4.8.1 can also be represented in this more generic form.

EXAMPLE 4 The expression '2018-34' is equivalent to the expression '2018-2G3MU' — the second quarter (second three-month period) of 2018.

5.4 Application within representations

5.4.1 General

Grouped time scale units can be applied to date and time representations as replacements for any time scale unit components as long as the resulting expression is a valid one.

EXAMPLE 1 '2018Y4G60DU6D' is a valid date expression that refers to the date year 2018, 6th day of the 4th 60-day group, which is the 186th ordinal day of year (in the Gregorian calendar, the 186th ordinal day is July 5th in a common year and July 6th in a leap year).

EXAMPLE 2 '2018Y9M2DT3GT8HU0H30M' is a valid date and time expression that refers to the date year 2018, September 2nd, 16:30 (30 minutes into the third 8-hour period). While the inner expression '0H' is supplied, it can be omitted in accordance with 7.10.

EXAMPLE 3 '2018-02-2G14DU' (implicit form) and '2018Y2M2G14DU' (explicit form) are complete date expressions that indicate "the second fortnight of February 2018", as specified by the rule "group every 14 days, second group".

EXAMPLE 4 '2018-03-3G10DU' (implicit form) and '2018Y3M3G10DU' (explicit form) are complete date expressions that indicate "the third 10-day block of March in 2018", as specified by the rule "group every 10 days, third group". A 10-day block is a time grouping commonly used in East Asia.

EXAMPLE 5 '16:1GT15MU' (implicit form) and 'T16H1GT15MU' (explicit form) are time expressions that indicate "the first quarter-hour of 16:00", as specified by the rule "group every 15 minutes, first group".

EXAMPLE 6 '2018-1G6MU' (implicit form) and '2018Y1G6MU' (explicit form) are date expressions of six-monthly precision that indicate "the first half of 2018", as specified by the rule "group every 6 months, first group", and therefore the grouping is specified in the [month] field. The resulting time interval is equivalent to the time interval expression '2018-01-01/2018-06-30' and the sub-year grouping expression '2018-40'.

5.4.2 Use of grouped units

Date and time representations can incorporate the grouped time scale units by applying rules from 5.2 and 5.3 to a date and time representation described within this document and [ISO 8601-1](#), by replacing a time scale unit component in the representations with [group].

The following representation is used for date and time representations (complete representation, reduced precision) that incorporate grouped time scale units. This representation is denoted as [groupDateTime] in this document.

groupDateTime = [higherOrderUnits][group][lowerOrderUnits]

where

[higherOrderUnits] contains time scale units (and their values) that are of higher order compared to the highest order time scale unit within the group unit;

[lowerOrderUnits] is optional, it contains time scale units (and their values) that are of equal or lower order compared to the lowest order time scale unit within the group unit.

EXAMPLE 1 '[monthE][group][yearE]' is an invalid date and time representation as [monthE] has a lower order than that of [yearE].

When [lowerOrderUnits] is omitted, the expression refers to a unit of the grouped time scale unit and therefore has the precision of the grouped time scale unit.

EXAMPLE 2 '[centE][group]' is a valid date and time representation, expressed in units of the grouped time scale unit. '10C5G20YU' is a sample expression of this representation that points to the fifth 20-year unit in the 10th century.

EXAMPLE 3 '1933Y1G80DU' represents the first 80-day group of the year 1933, with a precision of 80-days.

EXAMPLE 4 '1543Y1M3G5DU' refers to the third five-day group in January of the year 1543, with a precision of five-days.

EXAMPLE 5 '110Y2G3MU' refers to the second three-month block of the year 110, which is effectively referring to the second quarter of the year 110, with a precision of three-months.

EXAMPLE 6 '6GT2HU' refers to the sixth two-hour block, with a precision of two-hours.

When [lowerOrderUnits] is present, the expression refers to a date and time using the lowest time scale unit in the [lowerOrderUnits] expression, and therefore has the precision of that time scale.

Time scale components placed to the right of a grouped time scale unit behave identically to those after a typical time scale component, such that the component to the right refers to an instance or time interval "within" the component at the left.

EXAMPLE 7 '[yearE][group][dayE]' is a valid date and time representation, expressed in the calendar day time scale unit. The '[dayE]' points to a calendar day within the expression '[yearE][group]'. '2018Y2G3MU50D' is a sample expression of this representation that points to the 50th day of the second quarter of the year 2018.

EXAMPLE 8 '[decE][group][timeE]' is a valid date and time representation, expressed in the lowest order time scale units specified in [timeE], which is the clock second time scale unit. The '[timeE]' points to a clock time within an expression of '[decE][group]'. '201J2G5YU3DT10H0S' is a sample expression of this representation that includes all the 10:00:00 clock times of the third day of each year of the second five-year block in the 201th decade (spanning years 2010 to 2019). This can be expanded into the time interval set '{ 2015Y3DT10H0S, 2016Y3DT10H0S, 2017Y3DT10H0S, 2018Y3DT10H0S, 2019Y3DT10H0S }'.

EXAMPLE 9 '2018Y3G60DU6D' represents the sixth day within the third 60-day group of the year 2018.

EXAMPLE 10 '2018Y20GT12HU3H' represents the third hour within the twentieth 12-hour group of the year 2018.

5.4.3 Adherence to grouped unit boundaries

Time scale unit values provided in [lowerOrderUnits] shall not fall outside of the value bounds set by the group time scale unit (the [group] expression).

EXAMPLE 1 '2018Y1G2MU30D' refers to the thirtieth day of the first two-month group in the year 2018. Since the first two-month group of 2018 contains 59 days, the subsequent expression '30D' falls within the same grouped time scale unit.

EXAMPLE 2 '2018Y1G2MU60D' refers to the sixtieth day of the first two-month group in the year 2018. It is an invalid expression as the first two-month group of 2018 only contains 59 days, the expression '60D' falls outside of the same grouped time scale unit.

5.4.4 Representation with time shift

A representation using grouped time scale units can be represented with a time shift as:

[groupDateTime][shiftE]

EXAMPLE 1 '2018Y1G60DUZ-5H' represents the first 60-day group of the year 2018, five hours behind UTC.

EXAMPLE 2 '2018Y3G60DU6DZ8H' represents the sixth day of the third 60-day group of the year 2018, eight hours ahead of UTC.

5.4.5 Conversion to basic time scale units

5.4.5.1 General

When an expression referring to a grouped time scale unit has to be converted to use basic time scale units (time scale units that are defined in [4.3](#) and [ISO 8601-1:2019](#), 4.3), it can be converted into a time interval.

EXAMPLE '2018Y1G6MU' refers to the first 6-month group of year 2018. The resulting time interval is equivalent to that of the time interval expression '2018Y1M/2018Y6M'.

5.4.5.2 Truncation of partial units

When the duration of a grouped time scale unit does not cleanly divide the original time scale component, the last unit of the grouped time scale is only partially within the boundaries specified by the original time scale component.

In this case, the particular grouped time scale unit refers only to the duration that is within bounds of the original time scale component. The remaining duration that are out of bounds shall be truncated.

EXAMPLE '2018Y9M4G8DU' refers to the "fourth 8-day grouping of September of the year 2018", but the fourth grouped unit only has 6 days within September (2 days are in October). According to the above truncation rule, only the first 6 days belong to this grouped unit, and the resulting time interval is '2018-09-25/2018-09-30'.

5.4.5.3 Spanning grouped unit boundaries

In order to express a time interval covered by the grouped unit without regard to the boundaries of the original time component, the same expression can be provided using time interval representation.

EXAMPLE For '2018Y9M4G8DU' ("fourth 8-day grouping of September of the year 2018"), the start date of the fourth grouped unit can be found as '2018Y9M25D'. Therefore, the expression of '2018Y9M25D/P8D' provides the full 8-day duration of the unit, in the interval '2018Y9M25D/2018Y10M2D'.

6 Set representation

6.1 Set of date and time expressions

Curly braces enclosing a set of date and time expressions (with no specified order and separated by commas, with no intervening spaces) are used to mean "all members of the set".

NOTE Spaces occurring in set representation examples are for clarity only. There should be no spaces within the actual expressions.

`["{"}[expr1][","][expr2] ... ["}"]`

EXAMPLE 1 '{1960, 1961, 1962, 1963}' could be used to indicate the calendar years of publication of a book that was published in each of these calendar years: 1960, 1961, 1962 and 1963.

Expressions contained within a set may be of different precisions.

EXAMPLE 2 '{1960, 1961-12}' indicates the calendar year 1960 and the month December of 1961.

6.2 Single element amongst set

Square braces enclosing a set of date and time expressions (with no specified order and separated by commas, with no intervening spaces) are used to mean "one member of the set".

`["["][expr1][","][expr2] ... ["]"]`

The double brackets shown in this representation is a side effect of the usage of square brackets to separate description from representation. Square brackets shown in this representation should not occur in an expression except for the instances that were double-quoted as actual characters.

EXAMPLE 1 '[1984, 1986, 1988]' indicates that one of the calendar years 1984, 1986 or 1988 applies. This could be used to express "year of publication" when it is known that the desired calendar year is a calendar year within the given set.

EXAMPLE 2 '[1667, 1760-12]' indicates the calendar year 1667 or the calendar month December of 1760.

6.3 Range element expansion

The double-dot notation ("..") allows indication of "on or before", "on or after" of date and time expressions, and also of an expansion of a range between and including two date and time expressions.

The precision of the resulting expression (and the expanded elements) follows the date and time expression to which the double-dot notation applies. In the case where the double-dot indicates a range between two date and time expressions, both expressions should be of the same precision.

a) On or before a date and time expression, inclusive:

`[".."][expr]`

EXAMPLE 1 '..1984' represents the calendar year 1984 or a year before 1984.

EXAMPLE 2 '..1984-10' represents the calendar month October of 1984 or an earlier calendar month.

EXAMPLE 3 '..1760-12-03' represents December 3, 1760 or some earlier calendar day.

b) On or after a date and time expression, inclusive:

`[expr][".."]`

EXAMPLE 4 '1984..' represents the calendar year 1984 or a calendar year after 1984.

EXAMPLE 5 '1760-12..' represents the calendar month December 1760 or a later calendar month.

EXAMPLE 6 '1984-10-10..' represents the calendar day 10th of October 1984 or a calendar day after 10th of October 1984.

- c) All elements between two date and time expressions, inclusive:

[expr1][".."][expr2]

EXAMPLE 7 '1670..1673' represents the expansion of 1670, 1671, 1672 and 1673.

EXAMPLE 8 '1984-10-10..1984-11-01' represents the expansion of days 10th of October 1984 through 1st of November 1984.

- d) Any combination of a), b) and c)

EXAMPLE 9 '..1983-12-31, 1984-10-10..1984-11-01, 1984-11-05..' represents the days 31st of December 1983; 10th of October 1984 through 1st of November 1984; and 5th of November 1984 or any day after.

6.4 Set representations and expansion

Expansion described in [6.3](#) can be applied to representations described in [6.1](#) and [6.2](#).

EXAMPLE 1 '{1667, 1668, 1670..1672}' is expanded into the expression '{1667, 1668, 1670, 1671, 1672}', which indicates the calendar years of 1667, 1668, 1670, 1671 and 1672.

EXAMPLE 2 '[1760-01, 1760-02, 1760-12..]' represents a calendar month: January or February of 1760, or December 1760, or a later calendar month.

EXAMPLE 3 '{1M2S..1M5S}' is expanded into the expression '{1M2S, 1M3S, 1M4S, 1M5S}', which is the set of time scale components expressing the duration "one minute, two seconds", "one minute, three seconds", "one minute, four seconds", and "one minute, five seconds".

EXAMPLE 4 '[1M2S, 1M3S]' represents a single element of the time scale components 1M2S and 1M3S.

6.5 Expressions with time scale components

Representations described in [6.1](#), [6.2](#) and [6.3 c\)](#) can also be used with time scale components themselves, which may form partial date and time expressions.

EXAMPLE 1 '{P1M2S..P1M5S}' is expanded into the expression '{P1M2S, P1M3S, P1M4S, P1M5S}', which is the set of time scale components expressing the duration "one minute, two seconds", "one minute, three seconds", "one minute, four seconds" and "one minute, five seconds".

EXAMPLE 2 '[P1M2S, P1M3S]' represents a single element of the time scale components P1M2S and P1M3S.

6.6 Integer expressions

Representations described in [6.1](#), [6.2](#) and [6.3 c\)](#) can also be used with integers (for use in [Clause 12](#)).

EXAMPLE 1 '{1, 2..5}' is expanded into the expression '{1, 2, 3, 4, 5}', which indicates all of the integers 1, 2, 3, 4 and 5.

EXAMPLE 2 '[1, 3, 5]' represents a single element chosen among the integers 1, 3 or 5.

These two representations ("all of the integers" and "single element") may be used within a representation calling for an integer, for example, [i].

EXAMPLE 3 '2018-{1,3,5}G2MU' expresses the first, third and fifth two-month periods of 2018, a shorter form of the set representation of '{ 2018-01-01/2018-02-28, 2018-05-01/2018-06-30, 2018-09-01/2018-10-31 }' that uses [ISO 8601-1](#) time intervals.

EXAMPLE 4 '2018-[2,4]G3MU' expresses the second or fourth three-month periods of 2018.

7 Explicit representation for date and time

7.1 General

In [ISO 8601-1](#), explicit format is used for representation of duration only. This clause extends [ISO 8601-1](#) to define explicit forms for date and time representations. These representations are based on the mechanisms provided in CalConnect CC 18011, Clause 6, "Explicit representations".

7.2 Date

7.2.1 General

The representations provided in this clause are collectively denoted as '[dateE]'.

7.2.2 Calendar date

The complete representation of a calendar date shall be as follows.

[yearE(m)][monthE][dayE]

EXAMPLE '1985Y4M12D', calendar year 1985, April 12th.

7.2.3 Ordinal date

The complete representation of an ordinal date shall be as follows.

[yearE(m)][dayoE]

EXAMPLE '1985Y102O', calendar year 1985, ordinal day 102 of year.

7.2.4 Week date

A complete representation of a week date shall be as follows.

[yearE(m)][weekE][daykE]

EXAMPLE '1985Y15W5K', calendar year 1985, Friday of the calendar week 15.

7.3 Time of day

7.3.1 Local time of day

Representations of local time of day as defined below make no provisions to prevent ambiguities in expressions that result from discontinuities in the local time scale (e.g. daylight-saving time).

When the need arises to prevent these ambiguities, the representations provided in [7.6](#) may prove useful.

A complete representation of local time of day shall be as follows. This representation shall be denoted as [timeE].

timeE = ["T"][timeUnits]

timeUnits = [hourE][minE][secE]

EXAMPLE 1 'T23H20M50S', a local time of day that expresses the fiftieth second of the twentieth minute of the twenty-third clock hour.

[timeE] may be expressed in reduced precision, where only [hourE] or [hourE] and [minE] are present.

EXAMPLE 2 'T23H20M', a local time of day that expresses the twentieth minute of the twenty-third clock hour, with clock minute precision.

7.3.2 Beginning of the day

The complete representation of the beginning of the day in explicit form shall be as follows.

'T0H0M0S'

Applying the rule of omitting zero-valued time scale components (7.10), it can also be represented as:

'T0S'

There is no representation of end of day. This prevents ambiguity of the overlapping end of day and the beginning of the next day.

7.4 Time shift

A time shift is used to represent the shift of local standard time against UTC.

The complete representation of time shift in explicit form is represented as follows and is denoted as [shiftE] within this document.

shiftE = ["Z"]["-"] [timeUnits]

where

["Z"] is the time shift designator;

["-"] the leading minus sign is only applied when the time shift expressed is behind UTC, it shall be omitted if the time shift is ahead of or equal to UTC;

[timeUnits] represents the actual time shift from UTC, which is optional and may be of reduced precision.

In accordance with 7.10, a zero-valued time scale component may be omitted in [timeUnits].

EXAMPLE 1 'Z-5H', the time shift is 5 hours behind UTC.

EXAMPLE 2 'Z8H', the time shift is 8 hours ahead of UTC.

EXAMPLE 3 'Z28H', the time shift is 28 hours ahead of UTC.

EXAMPLE 4 'Z6H0M', the time shift is six hours ahead of UTC, with minute precision.

EXAMPLE 5 'Z7H33M14S', the time shift is seven hours, thirty-three minutes and fourteen seconds ahead of UTC. This expression represents the approximate time shift of local solar time to UTC, on the first day of year at Central, Hong Kong, as according to the equation of time.

A single ["Z"] with the [timeUnits] portion empty, in accordance with the rules specified in 7.10, indicates that the time shift from UTC of day is zero and is functionally equivalent to the representation 'Z0H0M0S'.

["Z"]

EXAMPLE 6 'Z', UTC where the time shift is zero.

EXAMPLE 7 'Z0H0M', UTC where the time shift is zero, with minute precision.

EXAMPLE 8 'Z0S', UTC where the time shift is zero, with second precision, where zero-valued components are omitted according to 7.10.

7.5 Date with shift

An event with day precision occurring at local time is often bound to a time shift.

The complete representation for date with a time shift is as follows.

[dateE][shiftE]

EXAMPLE 1 '1985Y4M12DZ-5H', calendar year 1985, April 12th with a time shift of 5 hours behind UTC.

EXAMPLE 2 '2018Y9M12DZ8H', calendar year 2018, September 12th with a time shift of 8 hours ahead of UTC.

NOTE A publication submission window is often of day precision and bound to a local time shift.

7.6 Time of day with time shift

The complete representation for time of day with a time shift is as follows.

[timeE][shiftE]

EXAMPLE 1 'T23H20M50SZ', 23:20:50 at UTC of day.

EXAMPLE 2 'T23H20M50SZ-5H0M', 23:20:50 at local time, which is 5 hours behind UTC.

EXAMPLE 3 'T23H20M50SZ8H', 23:20:50 at local time, which is 8 hours ahead of UTC.

7.7 Date and time of day

7.7.1 General

A time can be expressed by combining a date expression (7.2) with a time of day expression (7.3).

The date part of a date and time expression shall be complete. The time provided may be of reduced precision (7.3.1).

The representations provided in this subclause are collectively denoted as [datetimeE].

7.7.2 Date and time only

The complete representation for date and time only is as follows.

[dateE][timeE]

EXAMPLE '1985Y4M12DT23H20M30S', calendar year 1985, April 12th, 23:20:30.

7.7.3 Date and time with shift

The complete representation for date with time and a time shift is as follows.

[dateE][timeE][shiftE]

EXAMPLE '1985Y4M12DT23H20M30SZ8H', calendar year 1985, April 12th, 23:20:30, 8 hours ahead of UTC.

7.8 Decade

The complete representation for decade is as follows.

[decE]

EXAMPLE 1 '188J', the decade 1880s, spanning years from 1880 to 1889.

EXAMPLE 2 '18J', the decade 180s, spanning years from 180 to 189.

7.9 Century

The complete representation for century is as follows.

[centE]

EXAMPLE 1 '13C', the century 1300s, spanning years from 1300 to 1399.

EXAMPLE 2 '3C', the century 300s, spanning years from 300 to 399.

7.10 Omission of zero-valued components

A time scale component within an explicit form that has a value of '0' may be omitted entirely with its corresponding designator, except when the omission of it affects the indication of precision.

In the representations given in this subclause, if a time scale component of a higher order (compared to the lowest order component specified in the representation) is omitted, it is assumed to have the value "0" as long as the time scale component accepts the value "0" (see [ISO 8601-1:2019](#), 4.3).

EXAMPLE 1 '2018Y8M8DT30M0SZ' indicates the date 8th August 2018 at time 00:30:00 in UTC time of day. Since the "hour" component is omitted and "seconds" included, the hour is assumed to be zero, and the expression has second precision. This is equivalent to the expression '2018Y08M08DT30M0SZ' in which zero padding is applied to calendar month and calendar day.

EXAMPLE 2 '1985Y4M15DT15H0M10S' and '1985Y4M15DT15H10S' can be both used to express the time 3:00:10 p.m. on April 15th, 1985.

EXAMPLE 3 '1988Y3M1DT2H0M0S' and '1988Y3M1DT2H' are unequal expressions; the former expression has second precision, and the latter expression has hour precision.

7.11 Indication of precision

The lowest denoted time scale component in a date and time representation indicates the precision level of the representation.

EXAMPLE 1 '1985Y4M' has calendar month precision.

EXAMPLE 2 '1985Y4M12DT2H' has clock hour precision.

EXAMPLE 3 '1985Y4M12DT30M' has clock minute precision, where the clock hour time scale component is a zero-valued omission.

7.12 Decimal fractions for time

A decimal fraction of hour, minute or second may be included in an expression in accordance with [ISO 8601-1:2019](#), 5.3.1.4.

EXAMPLE 1 '2018Y8M8DT0,5H' indicates the date 8th August 2018 at time 00:30 with minute precision, using the comma decimal symbol.

EXAMPLE 2 '2018Y8M8DT10H30.5M' indicates the date 8th August 2018 at time 10:30:30 with second precision, using the comma decimal symbol.

EXAMPLE 3 '2018Y8M8DT10H30M15,3S' indicates the date 8th August 2018 at time 10:30:15 plus 300 ms with sub-second precision, using the comma decimal symbol.

7.13 Representations other than complete

Any of the representations given in this subclause may be modified for reduced precision ([7.11](#)) and decimal representation ([7.12](#)).

EXAMPLE 1 '2018Y8M' indicates the date August 2018 with calendar month precision.

EXAMPLE 2 '2018Y8M8DT30M': indicates the date 8th August 2018 at time 00:30 with minute precision. Since the "hour" component is omitted, it is assumed to be zero because it is of a higher order than the lowest order component specified in the representation ("minutes"). The omitted "second" component is not assumed to be zero because it is of a lower order than the lowest order component.

7.14 Time intervals

7.14.1 General

A time interval in explicit form shall be expressed by a start and an end date. A solidus ["/"] is used as a separator to separate the two start and end dates. The complete representations of a time interval are collectively referred by the symbol [timeIntervalE].

- a) Start and end:

$$\text{tiseE} = [\text{dtE}]["/"][\text{dtE}]$$

- b) Start and duration:

$$\text{tisdE} = [\text{dtE}]["/"][\text{duration}]$$

- c) End and duration:

$$\text{tideE} = [\text{duration}]["/"][\text{dtE}]$$

In these representations, occurrences of [dtE] may be replaced with [dtsE] which allows the specification of time shift.

EXAMPLE '1985Y4M12DT23H20M50S/1985Y6M25DT10H30M0S', time interval beginning at 20 minutes and 50 seconds past 23 hours on 12 April 1985 local time of day and ending at 30 minutes past 10 hours on 25 June 1985 local time of day.

7.14.2 Time scale component order

For expression of a time interval by a start and an end, higher order time scale components may be omitted from the "end of time interval", provided that the resulting expression is unambiguous. In this case the omitted higher order components from the "start of time interval" expression apply.

EXAMPLE 1 '2018Y1M15D/2M20D' is equivalent to '2018Y1M15D/2018Y2M20D' as the expression '2M20D' unambiguously refers to the calendar month and calendar day components, and the higher order components can be inherited from the start date expression.

EXAMPLE 2 '2018Y1M15D/2M10D' is an invalid expression. If the higher order components were to be inherited from the start date expression, then the end date would be earlier than the start date.

7.14.3 Time shift indication

Representations for time zones and UTC included with the component preceding the separator shall be assumed to apply to the component following the separator, unless a corresponding alternative is included.

EXAMPLE '2018Y1M15DZ5Y0M/2018Y2M20D' is equivalent to '2018Y1M15DZ5Y0M/2018Y2M20DZ5Y0M' as the 'Z5H0M' time shift also applies to the expression after the separator.

7.15 Recurring time intervals

The use of explicit form may be used in recurring time intervals. The complete representation of a recurring time interval is given as follows.

$$["R"][n]["/"][\text{timeIntervalE}]$$

8 Qualification of date and time expressions

8.1 General

This clause applies the rules of [4.5](#) to date and time expressions given in explicit form and the "extended format" of implied form defined in [ISO 8601-1](#). Principles and representations in this clause are based on Library of Congress EDTF.

8.2 Principles

8.2.1 Complete qualification

Qualification of an entire date and time expression is specified by a qualification symbol occurring at the rightmost end of the expression.

8.2.2 Group qualification

Qualification of a group of components is specified by a qualifier symbol occurring immediately to the right of a time scale component, which denotes that the qualification applies to that component's value as well as any component values to the left of that component.

EXAMPLE 1 In a calendar date expression, if the qualifying symbol occurs immediately to the right of the calendar day, it applies to the calendar day, calendar month and calendar year (the entire date expression).

EXAMPLE 2 In a calendar date expression, if the qualifying symbol occurs to the right of the calendar month, it applies to the calendar month and calendar year.

EXAMPLE 3 In a calendar date expression, if the qualifying symbol occurs to the right of the calendar year, it applies to the calendar year only.

8.2.3 Individual qualification

Qualification of an individual component is specified by a qualifier symbol:

- in implicit form, occurring immediately to the left of the time scale component;
- in explicit form, occurring between the time scale component's value and its designating symbol.

8.2.4 Preferred representations for resolving ambiguity

In cases where multiple representations convey the same meaning, the preferred form is one with:

- fewer qualification characters;
- no redundant qualifications; and
- closer alignment and sorting properties among other unqualified dates.

These rules should be applied to a date and time expression to obtain its preferred form:

- a) complete qualification ([8.2.1](#)) is preferred over group qualification ([8.2.2](#)), which is preferred over individual qualification of time scale components ([8.2.3](#));

EXAMPLE 1 '2015-02-28?' is preferred over '2015-02?-?28' because complete qualification is preferred.

EXAMPLE 2 '2015-02?-28' is preferred over '?2015-?02-28' because group qualification is preferred.

- b) redundant qualifications should be removed;

NOTE 1 For instance, when an identical qualification on a group and an individual qualification both apply to a time scale component, the individual qualification is redundant.

EXAMPLE 3 '2015-02?-28' is preferred over '2015-?02?-28' because the first "?" in the latter form is redundant.

- c) Combine qualifications when possible.

NOTE 2 For instance, individual qualifications on adjacent time scale elements can be combined into an expression with a single group qualification.

EXAMPLE 4 '2015-02%-28' is preferred over '%2015-%02-28' because fewer qualification characters is preferred.

8.3 Time scale components allowing qualification

8.3.1 General

The time scale components from 4.3 and [ISO 8601-1:2019](#), 4.3.1 to 4.3.7 are extended using the rules of 4.5 to allow individual qualification (see 8.2.3).

8.3.2 Calendar year, left qualified:

- a) Implicit:

$\text{year}(q) = [!][Q][YYYY]$

EXAMPLE 1 '?1985', the calendar year 1985 is uncertain.

- b) Explicit:

$\text{yearE}(q) = [i][!][Q][Y]$

EXAMPLE 2 '2018?Y', the calendar year 2018 is uncertain.

8.3.3 Calendar month, left qualified

- a) Implicit:

$\text{month}(q) = [!][Q][MM]$

EXAMPLE 1 '?03', the calendar month March is uncertain.

- b) Explicit:

$\text{monthE}(q) = [i][!][Q][M]$

EXAMPLE 2 '8?M', the calendar month August is uncertain.

8.3.4 Calendar week of year, left qualified

- a) Implicit:

$\text{week}(q) = [!][Q][W][WW]$

EXAMPLE 1 '?W53', the calendar week 53 is uncertain.

- b) Explicit:

$\text{weekE}(q) = [i][!][Q][W]$

EXAMPLE 2 '20?W', the calendar week 20 is uncertain.

8.3.5 Calendar day of month, left qualified

a) Implicit:

$\text{day}(q) = [!][Q][DD]$

EXAMPLE 1 '?31', the calendar day 31 is uncertain.

b) Explicit:

$\text{dayE}(q) = [i][!][Q][\text{"D"}]$

EXAMPLE 2 '11?D', the calendar week 11 is uncertain.

8.3.6 Calendar day of week, left qualified

a) Implicit:

$\text{dayk}(q) = [!][Q][K]$

EXAMPLE 1 '?5', the calendar day of week Friday is uncertain.

b) Explicit:

$\text{daykE}(q) = [i][!][Q][\text{"K"}]$

EXAMPLE 2 '3?K', the calendar day of week Wednesday is uncertain.

8.3.7 Calendar day of year, left qualified

a) Implicit:

$\text{dayo}(q) = [!][Q][OOO]$

EXAMPLE 1 '?306', the 306th calendar day of year is uncertain.

b) Explicit:

$\text{dayoE}(q) = [i][!][Q][\text{"O"}]$

EXAMPLE 2 '88?O', the 88th calendar day of year is uncertain.

8.3.8 Clock hour, left qualified

a) Implicit:

$\text{hour}(q) = [!][Q][hh]$

EXAMPLE 1 '~12', approximately 12 clock hours past start of calendar day.

b) Explicit:

$\text{hourE}(q) = [i][!][Q][\text{"H"}]$

EXAMPLE 2 '6~H', approximately six hours.

8.3.9 Clock minute, left qualified

a) Implicit:

$\text{min}(q) = [!][Q][mm]$

EXAMPLE 1 '~30', approximately 30 clock minutes past start of clock hour.

- b) Explicit:

$\text{minE}(q) = [i][!][Q][\text{"M"}]$

EXAMPLE 2 '30~M', approximately 30 minutes.

8.3.10 Clock second, left qualified

Clock Second and seconds duration are represented as follows:

- a) Implicit:

$\text{sec}(q) = [!][Q][ss]$

EXAMPLE 1 '%40', uncertain and approximate, 40 clock seconds past start of clock minute.

- b) Explicit:

$\text{secE}(q) = [i][!][Q][\text{"S"}]$

EXAMPLE 2 '30%S', uncertain and approximate, 30 seconds.

8.3.11 Decade, left qualified

- a) Implicit:

$\text{dec}(q) = [!][Q][YYYY]$

EXAMPLE 1 '?196', the decade of 1960s is uncertain.

- b) Explicit:

$\text{decE}(q) = [i][!][Q][\text{"J"}]$

EXAMPLE 2 '133?J', the decade of 1330s is uncertain.

8.3.12 Century, left qualified

- a) Implicit:

$\text{cent}(q) = [!][Q][YY]$

EXAMPLE 1 '?16', the century of 1600s is uncertain.

- b) Explicit:

$\text{centE}(q) = [i][!][Q][\text{"C"}]$

EXAMPLE 2 '22?C', the century of 2200s is uncertain.

8.4 Calendar date representations with qualification

8.4.1 Complete representation of a calendar date

The complete representation of a calendar date corresponds to the representation in [7.2](#) and [ISO 8601-1:2019](#), 5.2.2.1, b).

- a) Implicit format:

$\text{dateI}(q) = [\text{dateI}][Q]$

EXAMPLE 1 '1985-04-12?', a date of 1985 April 12th, where the entire date is uncertain.

b) Explicit format:

dateE(q) = [dateE][Q]

EXAMPLE 2 '1985Y04M12D?', a date of 1985 April 12th, where the entire date is uncertain.

8.4.2 Representations of calendar dates with reduced precision

These representations correspond to the representations in [ISO 8601-1:2019](#), 5.2.2.2.

a) A specific calendar month

— Implicit:

[year]["-"][month][Q]

— Explicit:

[yearE][monthE][Q]

EXAMPLE 1 '1985-04?', implicit format, representing 1985 April with month precision, where the entire date is uncertain.

EXAMPLE 2 '1985Y04M?', explicit format, representing 1985 April with month precision, where the entire date is uncertain.

b) A specific calendar year

— Implicit:

[year][Q]

— Explicit:

[yearE][Q]

EXAMPLE 3 '1985~', implicit format, representing the year 1985 with year precision, where the year value is approximate.

EXAMPLE 4 '1985Y?', explicit format, representing the year 1985 with year precision, where the year value is uncertain.

c) A specific decade

— Implicit:

[dec][Q]

— Explicit:

[decE][Q]

EXAMPLE 5 '198%', implicit format, representing the 1980s with decade precision, where the decade value is uncertain and approximate.

EXAMPLE 6 '198j?', explicit format, representing the 1980s with decade precision, where the decade value is uncertain.

d) A specific century

— Implicit:

[cent][Q]

— Explicit:

[centE][Q]

EXAMPLE 7 '19%', implicit format, representing the 1900s (century) with century precision, where the century value is uncertain and approximate.

EXAMPLE 8 '19C~', explicit format, representing the 1900s (century) with century precision, where the century value is approximate.

NOTE A decade expresses decade precision, "approximate decade" means that "the decade is approximated to be the 1980s, but it might be (for example) the 1970s or 1990s." It does not mean the time interval beginning approximately at the beginning of and ending approximately at the end of the 1980s. The latter would be expressed as 1980~/1990~ and would have year precision.

8.4.3 Expanded representations of calendar dates

These representations correspond to the representations in [ISO 8601-1:2019](#), 5.2.2.3.

If, by agreement, expanded representations are used, the formats shall be as specified below. The interchange parties shall agree to the additional number of digits in the time scale component year, decade and century.

In the examples below, it has been agreed to expand the time scale component year with two digits.

a) A specific calendar day

— Implicit, extended format:

[±][year(6)]["-"][month]["-"][day][Q]

EXAMPLE 1 '+001985-04-12%', the date 1985 April 12th with the entire date uncertain and approximate.

b) A specific calendar month

— Implicit, extended format:

[±][year(6)]["-"][month][Q]

EXAMPLE 2 '+001985-04%', the date 1985 April with month precision, with the entire date uncertain and approximate.

c) A specific calendar year

— Implicit, basic format:

[±][year(6)][Q]

EXAMPLE 3 '+001985%', the year 1985 with year precision, with the year uncertain and approximate.

d) A specific calendar decade

— Implicit, basic format:

[±][dec(5)][Q]

EXAMPLE 4 '+00198%', the decade 1980s with decade precision, with the decade uncertain and approximate.

e) A specific calendar century

— Implicit, basic format:

[±][cent(4)][Q]

EXAMPLE 5 '+0019%', the century 1900s with century precision, with the century uncertain and approximate.

8.4.4 Qualification of a group of time scale components from the right

These representations are obtained by applying the rules of [8.2.2](#) to calendar date representations.

a) Applies to calendar day, calendar month and calendar year

— Implicit:

[year]["-"][month]["-"][day][Q]

— Explicit:

[yearE][monthE][dayE][Q]

EXAMPLE 1 '2004-06-11%', implicit form, an uncertain and approximate date of 2004 June 11th.

EXAMPLE 2 '2004Y06M11D%', explicit form, an uncertain and approximate date of 2004 June 11th.

b) Applies to calendar month and calendar year

— Implicit:

[year]["-"][month][Q]["-"][day]

— Explicit:

[yearE][monthE][Q][dayE]

EXAMPLE 3 '2004-06~-11', implicit form, a date of 2004 June 11th where the month and year are approximate.

EXAMPLE 4 '2004Y06M~11D', explicit form, a date of 2004 June 11th where the month and year are approximate.

c) Applies to calendar year only

— Implicit:

[year][Q]["-"][month]["-"][day]

— Explicit:

[yearE][Q][monthE][dayE]

EXAMPLE 5 '2004?-06-11', implicit form, a date of 2004 June 11th where the year is uncertain.

EXAMPLE 6 '2004Y?06M11D', explicit form, a date of 2004 June 11th where the year is uncertain.

8.4.5 Qualification of individual time scale components

This representation is obtained by applying the rules of [8.2.3](#) to calendar date representations.

a) Implicit:

[year(q)]["-"][month(q)]["-"][day(q)]

b) Explicit:

[yearE(q)][monthE(q)][dayE(q)]

EXAMPLE 1 '2004-%06-%11', implicit form, '2004Y6%M11%D', explicit form: calendar year known; calendar month and calendar day uncertain and approximate.

EXAMPLE 2 '?2004-?06-11', implicit form, '2004?Y6?M11D', explicit form: calendar year and calendar month approximate; calendar day known.

EXAMPLE 3 '2004-?06-11', implicit form, '2004Y6?M11D', explicit form: uncertain calendar month; calendar year and calendar day known.

EXAMPLE 4 '2004-~06-11', implicit form, '2004Y6~M11D', explicit form: approximate calendar month; calendar year and calendar day known.

EXAMPLE 5 '2004-%06-11', implicit form, '2004Y6%M11D', explicit form: month uncertain and approximate; year and day known.

EXAMPLE 6 '?2004-06-?11', implicit form, '2004?Y6M11?D', explicit form: uncertain calendar year and calendar day; calendar month known.

EXAMPLE 7 '?2004-06-~11', implicit form, '2004?Y6M11~D', explicit form: calendar year uncertain; calendar month known; calendar day approximate.

EXAMPLE 8 '2004?-06-11', implicit form, '2004Y?6M11D', explicit form: uncertain calendar year; calendar month and calendar day known.

8.4.6 Allowing group and individual qualifications of time scale components

The rules of [8.2.2](#) and [8.2.3](#) are applied to calendar date representations.

EXAMPLE 1 '2004-06?-~11', implicit form, '2004Y6M?11~D', explicit form: calendar year and calendar month uncertain; calendar day approximate.

EXAMPLE 2 '2004?-06-~11', implicit form, '2004Y?6M11~D', explicit form: calendar year uncertain; month calendar known; calendar day approximate.

EXAMPLE 3 '2004-06%-?11', implicit form, '2004Y6M%11?D', explicit form: calendar year and calendar month uncertain and approximate; calendar day uncertain.

8.5 Date and time expressions with qualification

The principles in [8.2](#) are applied to date and time expressions using the time scale components described in [8.3](#).

EXAMPLE 1 '2018Y6M15DT19H20M30S~', explicit form, complete qualification: entire expression approximate.

EXAMPLE 2 '2018Y6M15D?T19H20M30S', explicit form, group qualification: calendar year, calendar month and calendar day uncertain; clock hour, clock minute and clock seconds known.

EXAMPLE 3 '2018Y6M?15DT19H20M30~S', explicit form, group and individual qualification: calendar year and calendar month uncertain; clock second approximate; calendar day, clock hour and clock minute known.

9 Unspecified digits

9.1 General

This clause applies the rules specified in [4.6](#) of unspecified digits to date and time expressions. The mechanisms and representations in this clause originate from Library of Congress EDTF.

9.2 Calendar date representations with unspecified digits

9.2.1 Unspecified time scale component values from the right

9.2.1.1 Within complete representations

The unspecified value character can substitute one or all digits of the right-most time scale component value in a date and time representation.

- a) Complete representation, calendar year and calendar month specified, calendar day unspecified

— Implicit

[year]["-"][month]["-"]["XX"]

— Explicit

[yearE][monthE]["X*D"]

EXAMPLE 1 '1985-04-XX' and '1985Y04MX*D' both indicate a calendar day in the calendar month of 1985 April, with the actual day unspecified, in implicit form and explicit form respectively.

- b) Complete representation, calendar year specified, calendar month and calendar day unspecified

— Implicit

[year]["-"]["XX"]["-"]["XX"]

— Explicit

[yearE]["X*M"]["X*D"]

EXAMPLE 2 '1985-XX-XX' and '1985YX*MX*D' both indicate a calendar day in the calendar year of 1985, with the actual month and day unspecified, in implicit form and explicit form respectively.

- c) Complete representation, calendar year, calendar month and calendar day unspecified

— Implicit

["XXXX"]["-"]["XX"]["-"]["XX"]

— Explicit

["X*Y"]["X*M"]["X*D"]

EXAMPLE 3 'XXXX-XX-XX' and 'X*YX*MX*D' both indicate a calendar date that has the actual year, month and day unspecified, in implicit form and explicit form respectively.

9.2.1.2 Within reduced precision dates

The unspecified value character can substitute all digits of the right-most time scale component value in the following cases for reduced precision dates:

- a) Reduced precision, calendar year specified, calendar month unspecified

— Implicit

[year]["-"]["XX"]

— Explicit

[yearE]["X*M"]

EXAMPLE 1 '2004-XX' and '2004YX*M' both indicate a calendar date with month precision in the year 2004 with the actual month unspecified, in implicit form and explicit form respectively.

b) Reduced precision, calendar year and calendar month unspecified

— Implicit

["XXXX"]["-"]["XX"]

— Explicit

["X*Y"]["X*M"]

EXAMPLE 2 'XXXX-XX' and 'X*YX*M' both indicate a calendar date with year precision with the actual year and month unspecified, in implicit form and explicit form respectively.

c) Reduced precision, one to four rightmost calendar year digits unspecified

1) One rightmost calendar year digit unspecified

— Implicit

[YYY]["X"]

— Explicit

[YYY]["X"]["Y"]

EXAMPLE 3 '201X' and '201XY' both indicate a calendar date with year precision, where the year number has four digits and starts with '201', in implicit form and explicit form respectively.

2) Two rightmost calendar year digits unspecified

— Implicit

[YY]["XX"]

— Explicit

[YY]["XX"]["Y"]

EXAMPLE 4 '20XX' and '20XXY' both indicate a calendar date with year precision, where the year number has four digits and starts with '20', in implicit form and explicit form respectively.

3) Three rightmost calendar year digits unspecified

— Implicit

[Y]["XXX"]

— Explicit

[Y]["XXX"]["Y"]

EXAMPLE 5 '2XXX' and '2XXXY' both indicate a calendar date with year precision, where the year number has four digits and starts with '2', in implicit form and explicit form respectively.

4) Four rightmost calendar year digits unspecified

— Implicit

["XXXX"]

— Explicit

["XXXX"]["Y"]

EXAMPLE 6 'XXXX' and 'XXXXY' both indicate a calendar date with year precision, where the year number is unspecified but with four digits, in implicit form and explicit form respectively.

d) Reduced precision, all calendar year digits unspecified

— Explicit

["X*"]["Y"]

EXAMPLE 7 'X*Y' indicates a calendar date with year precision, where the year number is unspecified with no digit limit, in explicit form.

NOTE The expression '1985-XX-XX' refers to a day in the year 1985 and differs in meaning from the expression '1985', which refers to the year 1985. Similarly, '1985-04' is an expression with reduced precision, while '1985-04-XX' is an expression of the complete representation.

9.2.2 Unspecified digit anywhere in time scale component

This clause adopts the rules specified in [4.6.3](#) for calendar date representations.

EXAMPLE 1 '156X-12-25', implicit form, '156XY12M25D', explicit form: December 25 sometime during the 1560s.

EXAMPLE 2 '15XX-12-25', implicit form, '15XXY12M25D', explicit form: December 25 sometime during the 1500s.

EXAMPLE 3 '1XXX-12-XX', implicit form, '1XXXY12MX*D', explicit form: some calendar day in December in some year during the century of 1000.

EXAMPLE 4 'XXXX-12-XX', implicit form, 'X*Y12MX*D', explicit form: some calendar day in December in some year.

EXAMPLE 5 '1560-XX-25', implicit form, '1560YX*M25D', explicit form: the 25th day of some calendar month in year 1560.

EXAMPLE 6 '1560-X2', implicit form, '1560YX2M', explicit form: the calendar month of either February or December of the year 1560.

EXAMPLE 7 '1XXX-XX', implicit form, '1XXXYX*M', explicit form: some calendar month during the 1000s.

EXAMPLE 8 '1XXX-12', implicit form, '1XXXY12M', explicit form: some December during the 1000s.

EXAMPLE 9 '1XXX', implicit form, '1XXXY', explicit form: some year during the 1000s.

EXAMPLE 10 '1XX3', implicit form, '1XX3Y', explicit form: some year ending in 3 during the 1000s.

9.3 Date and time expressions with unspecified digits

The rules in [4.6](#) are applied to any date and time expression.

EXAMPLE 1 '2018Y3M17DT20H40MX*S', explicit form, all components known except with clock second not specified.

EXAMPLE 2 '2018Y3M17DT20HXXMXS', explicit form, all components known, except that clock minute is a two-digit unspecified value, and clock second a one-digit unspecified value.

EXAMPLE 3 '201XY3M17DTX*H1XXMX0S', explicit form, unspecified calendar year between 2010 and 2019, March 17th, unspecified clock hour, unspecified two-digit clock minute, and a two-digit clock second with the first digit unspecified where the second digit is 0.

10 Extended time interval representations

10.1 General

The following features are used in conjunction with time intervals defined in [ISO 8601-1](#). The mechanisms and representations within this clause are based on features provided in Library of Congress EDTF.

10.2 Unknown or open start or end time intervals

The following representations extend the specification of time interval in [ISO 8601-1](#) to allow specification that the start or end date is unknown or open.

a) Open end time interval

[intexpr]["/"][".."]

EXAMPLE 1 '1985-04-12/..', implicit form, '1985Y4M12D/..', explicit form: the time interval starting at 1985 April 12th, with day precision with an open end.

EXAMPLE 2 '1985-04/..', implicit form, '1985Y4M/..', explicit form: the time interval starting at 1985 April, with month precision with an open end.

EXAMPLE 3 '1985/..', implicit form, '1985Y/..', explicit form: the time interval starting at year 1985, with year precision with an open end.

b) Open start time interval

[".."]["/"][intexpr]

EXAMPLE 4 '../1985-04-12', implicit form, '../1985Y4M12D', explicit form: the time interval with an open start and ending at 1985 April 12th, with day precision.

EXAMPLE 5 '../1985-04', implicit form, '../1985Y4M', explicit form: the time interval with an open start and ending at 1985 April, with month precision.

EXAMPLE 6 '../1985', implicit form, '../1985Y', explicit form: the time interval with an open start and ending at year 1985, with year precision.

c) Time interval with unknown end

[intexpr]["/"]

EXAMPLE 7 '1985-04-12/', implicit form, '1985Y4M12D/', explicit form: the time interval starting at 1985 April 12th, with day precision with an unknown end.

EXAMPLE 8 '1985-04/', implicit form, '1985Y4M/', explicit form: the time interval starting at 1985 April, with month precision with an unknown end.

EXAMPLE 9 '1985/', implicit form, '1985Y/', explicit form: the time interval starting at year 1985, with year precision with an unknown end.

d) Time interval with unknown start

["/"][intexpr]

EXAMPLE 10 '/1985-04-12', implicit form, '/1985Y4M12D', explicit form: the time interval with an unknown start and ending at 1985 April 12th, with day precision.

EXAMPLE 11 '/1985-04', implicit form, '/1985Y4M', explicit form: the time interval with an unknown start and ending at 1985 April, with month precision.

EXAMPLE 12 '/1985', implicit form, '/1985Y', explicit form: time interval with an unknown start and ending at year 1985, with year precision.

10.3 Qualification of dates in time intervals

10.3.1 General

The start and/or end date in an interval may be qualified.

10.3.2 Complete qualification

The start and end dates can be used in conjunction with usage specified in [4.5](#) and [4.6](#).

EXAMPLE 1 '1984-01-02~/2004-06-04', implicit form, '1984Y1M2D~/2004Y6M4D', explicit form: beginning approximately 1984-01-02 and ending 2004-06-04.

EXAMPLE 2 '1984~/2004-06', implicit form, '1984Y~/2004Y6M', explicit form: beginning approximately 1984 and ending June 2004.

EXAMPLE 3 '1984/2004-06~', implicit form, '1984Y/2004Y6M~', explicit form: beginning 1984 and ending approximately June 2004.

EXAMPLE 4 '1984~/2004%', implicit form, '1984Y~/2004Y%', explicit form: beginning is uncertain but thought to be 1984; end is uncertain but thought to be approximately 2004.

10.3.3 Partial qualification

Portions of the start and/or end dates may be qualified.

EXAMPLE '2004-06-~01/2004-06-~20', implicit form, '2004Y6M1~D/2004Y6M20~D', explicit form: a time interval in June 2004 beginning approximately the 1st and ending approximately the 20th.

10.4 Unspecified portions of dates in time intervals

Portions of the start or end date may be unspecified.

EXAMPLE '2004-06-XX/2004-07-03', implicit form, '2004Y6MXD/2004Y7M3D', explicit form: the time interval began on an unspecified day in June 2004 and ended July 3.

10.5 Uncertain and approximate dates in unknown or open time intervals

The start and end dates can be used in conjunction with usage specified in [4.5](#), [4.6](#) and [10.2](#).

EXAMPLE 1 '1984-01-02~/', implicit form, '1984Y1M2D~/', explicit form: beginning approximately January 2nd of year 1984; end date unknown.

EXAMPLE 2 '1984-01-02~/..', implicit form, '1984Y1M2D~/..', explicit form: beginning approximately 1984-01-02; no end date.

EXAMPLE 3 '../198X-01-02~', implicit form, '../198XY1M2D~', explicit form: beginning unknown; end date approximately January 2 of one of the years from 1980 to 1989.

10.6 Before and after with qualified time scale components

The start and end dates can be used in conjunction with the double dot notation [".."] specified in [6.3](#).

EXAMPLE 1 '..2004-06-01/2004-06-~20', implicit form, '..2004Y6M1D/2004Y6M20~D', explicit form: a time interval beginning before or on 1 June 2004 and ending in the same month, on approximately the 20th.

EXAMPLE 2 '2004-06-01~/2004-06-20..', implicit form, '2004Y6M1D~/2004Y6M20D..', explicit form: a time interval beginning approximately 1 June 2004 and ending the 20th or later.

11 Explicit duration and extensions

11.1 General

Duration can be expressed by a combination of time scale units in explicit form — years, months, weeks, days, hours, minutes and seconds. The representations of duration in this clause are based on CalConnect CC 18011, Clause 7, "Explicit duration".

These time scale components are specified in [4.3](#) and are used to specify positive duration.

11.2 Durational units

Time scale unit components are represented by a time scale component used to compose a duration expression and are defined as "durational units".

The following time scale components can be combined to form a representation denoted as [durationUnits] for the expression of duration.

$$\text{durationUnits} = [\text{yearE}][\text{monthE}][\text{weekE}][\text{dayE}][\text{"T"}][\text{hourE}][\text{minuteE}][\text{secondE}]$$

where

[**"T"**] is the time designator symbol used to separate the time scale components relating to the clock from those of the calendar;

each durational unit represented by a time scale component (such as [monthE]) may be omitted.

Within [durationUnits], time scale components of higher order shall be expressed before those of lower order.

Individual duration units are allowed to have negative values. The following representation denoted as [durationUnits(m)] accept negative values per component.

$$\text{durationUnits(m)} = [\text{yearE(m)}][\text{monthE(m)}][\text{weekE(m)}][\text{dayE(m)}][\text{"T"}][\text{hourE(m)}][\text{minuteE(m)}][\text{secondE(m)}]$$

11.3 Representations

11.3.1 General

The following representations are both considered complete representations of duration.

11.3.2 Composite representation

The composite representation of a duration is a more flexible and relaxed specification for duration than that of [ISO 8601-1:2019](#), 5.5.2. It accepts all expressions of the duration representation given in [ISO 8601-1:2019](#), 5.5.2 and is given as follows.

$$[!][\text{"-"}][\text{"P"}][\text{durationUnits(m)}]$$

where [durationUnits(m)] contains time scale components for expressing (positive or negative) duration (see [11.2](#)).

Expressions in the two examples below are valid in [ISO 8601-1](#).

EXAMPLE 1 'P3D', duration of three days.

EXAMPLE 2 'P180Y800D', duration of one-hundred-and-eighty years and eight-hundred days.

Expressions in the following four examples below are not valid in [ISO 8601-1](#), but are valid as specified in this clause.

EXAMPLE 3 'P3W2D', duration of three weeks and two days, which is 23 days (equivalent to the expression 'P23D'). In [ISO 8601-1](#), ["W"] is not permitted to occur along with any other component.

EXAMPLE 4 'P5Y10W', duration of five years and ten weeks.

EXAMPLE 5 'P-3M-3DT1H5M', duration of three months and three days in the reverse direction, with one hour and five minutes in the original direction.

EXAMPLE 6 'P-2M-1D', duration in the reverse direction of two months and one day.

When a minus sign is provided as prefix to the duration designator ["P"], the minus sign can be internalized into individual time scale components within the duration expression by applying to every time scale component within.

EXAMPLE 7 '-P2M1D' is equivalent to 'P-2M-1D'.

EXAMPLE 8 '-P5DT10H' is equivalent to 'P-5DT-10H'.

When a minus sign is applied to a time scale component whose value is already negative (pointing to the reverse direction), it means that the direction of duration should be once again reversed and should be turned into a positive value.

EXAMPLE 9 '-P8M-1D', duration in reverse, "eight months minus one day", is equivalent to 'P-8M1D', "eight months ago with a day ahead".

EXAMPLE 10 '-P-5WT-18H30M', duration in reverse, "go back five weeks, eighteen hours but thirty minutes ahead", is equivalent to 'P5WT18H-30M', "go ahead five weeks, eighteen hours, but thirty minutes back".

NOTE The exact duration for some time scale components can be known only when placed on the actual time scale, see [D.2](#).

11.3.3 Precedence representation

In resolution of date time formulas, the precedence order of time scale components for duration calculations may fundamentally alter the results.

The precedence representation of a duration, used for specifying author intent in the order of evaluation of time scale components within a duration, is described as follows, denoted as [durationP].

durationP = [durC-1][durC-2]...[durC-n]

where

[durC-i] is either ["P"][calendarDU] or ["P"]["T"][clockDU];

[calendarDU] is one of [yearE(m)], [monthE(m)], [weekE(m)] and [dayE(m)];

[clockDU] is one of [hourE(m)], [minute(m)] and [secondE(m)];

the order of the [durC-i] components determines the evaluation precedence order of the duration statement [durationP], from left to right.

The following four examples all specify a duration of one year, three months and two days, but the precedence order differs.

EXAMPLE 1 'P1YP3MP2D' describes a duration of one year, three months and two days, to be evaluated in the order from left to right.

EXAMPLE 2 'P2DP3MP1Y' describes a duration of two days, three months and one year, to be evaluated in the order from left to right.

EXAMPLE 3 'P3MP2DP1Y' describes a duration of three months, two days and one year, to be evaluated in the order from left to right.

EXAMPLE 4 'P-3MP2DP-1Y' describes a duration of three months in the reverse direction, two days in the original direction and one year in the reverse direction, to be evaluated in the order from left to right.

When a precedence duration is prefixed with a minus sign prior to the first ["P"], the minus sign can be internalized into all durational units within.

EXAMPLE 5 '-P2MP1D' is equivalent to 'P-2MP-1D'.

EXAMPLE 6 '-P5DPT10H' is equivalent to 'P-5DPT-10H'.

Similar to 11.3.2, when a minus sign is applied to a time scale component whose value is already negative, it should be turned positive.

EXAMPLE 7 '-P8M-1D', duration in reverse, "eight months minus one day", is equivalent to 'P-8MP1D', "eight months ago with a day ahead".

EXAMPLE 8 '-P-5WPT-18HPT30M', "go back five weeks, eighteen hours but thirty minutes ahead", is equivalent to 'P5WPT18HPT-30M', "go ahead five weeks, eighteen hours, but thirty minutes back".

The differences between composite representation and precedence representation are apparent during duration arithmetic, see [Clause 14](#).

11.4 Fractional duration

A duration representation can be modified to accept a fractional value in place of any [i] specified in a durational time scale component.

EXAMPLE 'P1M2.5D' expresses "one month and two and a half days".

12 Selection of date and time

12.1 General

A time scale component can be replaced with selection rules to specify matching criteria of certain time scale unit values. This clause builds upon the "explicit" syntax of time scale components specified in 4.3 and [ISO 8601-1:2019](#), 4.3. The principles and mechanisms of this clause originate from CalConnect CC 18012, Clause 5, "Selection of date and time".

The "selection rule set" [selection] may include zero or more selection rules.

selection = ["L"][selection-rule-1]...[selection-rule-n]["N"]

where

["L"] is the selection prefix designator;

["N"] is the selection suffix designator;

[selection-rule-n] is a selection rule defined in this cause; multiple selection rules may exist between the selection prefix and suffix designator.

A selection rule set expression resolves to a set of one or more time scale components, called a "selection set", which may be continuous or disjoint on a time scale.

EXAMPLE 1 'L{1,3,5}KN', "Mondays, Wednesdays and Fridays", describes a set of disjoint calendar days (all three calendar days are not adjacent to each other).

EXAMPLE 2 'L{1,2}MN', "Januarys and Februarys", describes a set of calendar months that are continuous on one side (January is next to February), but disjoint on the other, when placed on the time scale (March to December are excluded from this expression).

There can be only one selection rule of the same kind within a selection rule set.

NOTE 1 When a selection rule set incorporates a "selection of position" rule, the expression resolves to a single instance of a time scale component (or a discrete set of instances when the instance indicator, I, is expressed as a set).

EXAMPLE 3 'L1K1IN', "the first Monday", describes a single calendar day.

EXAMPLE 4 'L1K{1,2}IN', "the first and second Monday".

Selection rules shall be placed with higher-order time scale components on the left, and lower-order ones on the right.

NOTE 2 Selection rules specified in this clause are a superset to those of the "RECUR" syntax specified in RFC 5545; while it is possible to translate certain selection rules specified in this document to that syntax, exceptions and limitations are noted within context of the rules specified below.

12.2 Selection rules

12.2.1 Selection of calendar month of year

This selection rule specifies a set of calendar months of the calendar year.

Representation:

monthSR = [monthE]

Valid values are [1] to [12], corresponding to the ordinal number of the calendar month.

EXAMPLE 1 '3M' represents the third calendar month of year, i.e. March.

EXAMPLE 2 '12M' represents the twelfth calendar month of year, i.e. December.

12.2.2 Selection of calendar week of year

This selection rule specifies a set of calendar weeks of the calendar year. Since the calendar week of year component has a changeable bound, negative values of week are allowed for specifying the week of year number in reverse.

Representation:

weekSR = [weekE]

Valid values are [1] to [53] and [-53] to [-1]. This corresponds to the number of calendar weeks of a year according to week numbering as defined in [ISO 8601-1](#).

NOTE 1 RFC 5545 specifies that this rule should only be present when the frequency of the repeat rule is set to yearly ([13.4.2](#)).

EXAMPLE 1 The tenth week of the calendar year is represented by the expression '10W'.

EXAMPLE 2 The second last week of the calendar year is represented by the expression '-2W'.

NOTE 2 Week 53 can occur only when Thursday is January 1 or if it is a leap calendar year and Wednesday is January 1, in accordance with [ISO 8601-1:2019](#), 4.2.2.

NOTE 3 Refer to [4.4.1.3](#) for negative values of weeks.

NOTE 4 Definitions of the calendar week and the week number are provided in [ISO 8601-1](#).

12.3 Selection of calendar day of month

This selection rule specifies a set of days of the calendar month. Since the calendar day of month has a changeable bound, negative values of calendar day of month are allowed for specifying the day of month ordinal number in reverse.

Representation:

daySR = [dayE]

Valid values are [1] to [31] and [-31] to [-1]. This corresponds to the maximum number of calendar days of a calendar month.

EXAMPLE 1 '18D' represents the eighteenth calendar day of the calendar month.

EXAMPLE 2 '-10D' represents the tenth to the last calendar day of the calendar month.

NOTE 1 RFC 5545 does not allow this selection rule when the frequency part is set to weekly.

NOTE 2 Refer to [4.4.1.4](#) for negative values of calendar days of month.

12.4 Selection of week days

This selection rule specifies a set of days of the week.

Representation:

daykSR = [daykE]

Valid values are [1] to [7].

EXAMPLE 1 With a monthly repeat rule (i.e. when [eligible-time-intervals] is [monthE]; see [D.4.1](#)), '1K' represents all Mondays within a calendar month.

EXAMPLE 2 In a yearly context, '5K' represents all Fridays in the calendar year.

12.5 Selection of ordinal days in calendar year

This selection rule specifies a set of ordinal days of the calendar year and should only be specified when the repeat rule is set to yearly, monthly or daily ([13.4.2](#)).

Since the number of days in a calendar year has a changeable bound, negative values of calendar day of year are allowed for specifying the day of year ordinal number in reverse.

Representation:

dayoSR = [dayoE(m)]

Valid values are [1] to [366] and [-366] to [-1].

NOTE The values of [366] and [-366] are used to match a calendar leap year.

EXAMPLE 1 '-10' represents the last day of the calendar year (December 31st).

EXAMPLE 2 '-3070' represents the 307th to the last day of the calendar year (February 28th of a common year).

12.6 Selection of hours

This selection rule specifies a set of hours of the calendar day.

Representation:

hourSR = [hourE]

Valid values are [0] to [23].

EXAMPLE 1 '23H' represents hour 23 (the last hour) of a day.

EXAMPLE 2 '0H' represents the beginning hour of day.

12.7 Selection of minutes

This selection rule specifies a set of minutes within an hour.

Representation:

minSR = [minE]

Valid values are [0] to [59].

EXAMPLE 1 '59M' represents minute 59 (the last minute) of an hour.

EXAMPLE 2 '3M' represents minute three of an hour.

12.8 Selection of seconds

This selection rule specifies a set of seconds within a minute.

Representation:

secSR = [secE]

Valid values are [0] to [60].

EXAMPLE '30S' represents second 30 of the clock hour.

NOTE 1 The value of [60] is used to match a leap second.

NOTE 2 RFC 5545 specifies that the value of [60] should be changed to [59] on conversion according to the rule of "BYSECOND" since it does not support a value of [60].

12.9 Selection of position

The position rule, if present, should be applied last, and only when there is at least one selection rule preceding it. It selects a set of occurrences that is a subset of the occurrences already selected (by the selection rules preceding it).

positionSR = [position]["I"]

where [position] is a positive [i] or negative integer [negi].

When used in conjunction with a set of occurrences, [position] identifies the i-th occurrence amongst the set of occurrences.

Specifically, position numbers within a set of occurrences are considered to start with [1] (the first occurrence of the set of occurrences), and [-1] represents the last occurrence, [-2] the next-to-last, and so on.

EXAMPLE 1 With a monthly repeat rule, "the last work day of the calendar month" can be represented as the selection rule 'L{1..31}D{1..5}K-1IN'; where '{1..31}D' represents all days of all months, '{1..5}K' represents Mondays to Fridays, '-1I' represents the last instance.

EXAMPLE 2 With a yearly repeat rule, the selection rule 'L{1..7}O1K1IN' expresses "the first Monday of the calendar year". With a monthly rule it expresses "the first Monday of the calendar month". '{1..7}O' represents the first seven days of the calendar year/month, '1K' represents Monday, '1I' represents the first instance.

EXAMPLE 3 With a yearly repeat rule, "the second Monday of the calendar year" can be expressed as the selection rule 'L{1..14}O1K2IN'.

Valid values for [position] are bound to the time scale components in the context. The integer value [0] is not allowed.

EXAMPLE 4 With a monthly repeat rule, the expression 'L{1,2,3}D100IN' is invalid because in a single calendar month, there are only three days that can match the calendar day of month numbers 1, 2 and 3, therefore the position of 100 cannot be selected.

Time scale components of a lower order than that of the resulting selection set can be used to further refine the selection set, identical to using a lower order time scale component to refine a higher order time scale component, such as by specifying clock minute to refine the clock hour.

EXAMPLE 5 The expression 'L2K2INT9H' refers to 9 a.m. on the second Tuesday.

12.10 Selection with time interval

This selection rule allows extending a selection set with a duration, which applies to each of the elements in the selection set.

Each element of the selection set becomes the start of an interval whose duration is as specified by the following representation:

timeIntervalSR = [selection]["/"][duration]

where

[selection] is a selection rule set specified in [12.2](#), each element of which becomes the start of a time interval;

[duration] is a valid duration that defines the duration of each time interval.

EXAMPLE 1 'LL3K4IN/P5DN' selects a time interval set with the start on "the fourth Wednesday" for a duration of five days, where 'L3K4IN/P5D' is a selection with time interval.

The representation of this selection rule effectively allows nesting of selection rules by embedding a selection rule set within. Boundaries of nested selection rule sets shall be set by pairs of the selection prefix (["L"]) and suffix (["N"]) from the outside-in, where the outermost pair indicate the outermost selection rule set, and so on.

EXAMPLE 2 'LL1K{1,3}IN/P5DN' expresses the selection of a set of two time intervals starting on the first and third Mondays, each of a duration of 5 days. There are two selection rule sets in this expression. The innermost selection rule set is 'L1K{1,3}IN', which describes "first and third Monday". The duration statement of 'P5D' is then applied to that selection set effectively turning each element into the start of a time interval with the duration of 5 days.

EXAMPLE 3 'LLL2K2IN/P10DN4K2IN' expresses the selection of "the second Thursday following the second Tuesday". The innermost selection set, 'L2K2IN', expresses "second Tuesday". '/P9D' is then applied, resulting in "interval beginning the second Tuesday for duration 10 days". '4K2I', "second Thursday" is then applied resulting in "second Thursday within the ten-day block beginning the second Tuesday", effectively, "second Thursday following the second Tuesday".

12.11 Application within representations

12.11.1 General

Selection rules may be used with any date and time representation in explicit form. This representation is especially useful for describing movable days.

EXAMPLE 1 '2018Y3ML1K1IN' is a valid date expression that refers to the date year 2018, first Monday of March.

EXAMPLE 2 '2018Y9MTLT8H20MN3I' is a valid date and time expression that refers to the date year 2018, September, the third instance of 08:20 (which is September 3rd, 08:20).

EXAMPLE 3 '{2018,2019,2020,2021,2022}YL2M29D1IN' is a valid date expression that selects the February 29 leap day. It matches the first instance of the date February 29th in the years 2018 to 2022.

The following examples are used in conjunction with rules specified in [4.6.2](#).

EXAMPLE 4 'L5M7K2IN' is an expression for "the second Sunday of May". When placed in an expression with yearly context, such as in 'X*YL5M7K2IN', the date falls on the International Mother's Day. ('X*Y' expresses "unspecified year"; see [4.6](#)).

EXAMPLE 5 'L11M4K4INT17HZ-5H' is an expression for the date "5:00:00 p.m. of the fourth Thursday in November, in UTC-05:00". When placed in an expression with yearly context, such as in 'X*YL11M4K4INT17HZ-5H', it represents 5:00:00 p.m. falling on the U.S. Thanksgiving Day.

EXAMPLE 6 'L4M{19..26}D4K1IN' is an expression for the first Thursday after April 18th. When placed in an expression with yearly context, such as in 'X*YL4M{19..26}D4K1IN', the date falls on the First Day of Summer (*sumardagurinn fyrsti*) in Iceland.

EXAMPLE 7 'LL4M4D/-P20DN7K-2IN' is an expression for the second Sunday before April 4th. April 4th is the earliest candidate day of the fifth solar term of the traditional East Asian lunisolar calendar, which is traditionally observed in East Asia as the Ching Ming Festival (the Qingming Festival, Tomb-Sweeping Day or Ancestor's Day). This expression can be used to schedule preparation of traditional rituals two Sundays prior to the festival.

EXAMPLE 8 'XXX{0,2,4,6,8}Y11MLLL1K1IN/P9DN2K1IN' is an expression for the selection of "the first Tuesday following the first Monday of November of any four-digit even numbered year." This is Election Day in the United States.

12.11.2 Context set by selection

A selection expression can be used to describe one or more dates or time intervals. Time scale components can be placed after a selection in which case they apply to all the dates and intervals described.

EXAMPLE 1 '2018YL1K1IN' is an expression for a specific calendar day. An expression with time scale components immediately after, such as '2018YL1K1INT10H0M0S', represents a specific date and time.

EXAMPLE 2 '2018YL{1,2,5}KN' is an expression for a set of days (every Monday, Tuesday and Friday of 2018). An expression with time scale components immediately after, such as '2018YL{1,2,5}KNT10H0M0S' (every Monday, Tuesday and Friday of 2018 at 10AM) describe a time interval set '{ 2018Y1M1DT10H0M0S, 2018Y1M2DT10H0M0S, 2018Y1M5DT10H0M0S, 2018Y1M8DT10H0M0S, 2018Y1M9DT10H0M0S, 2018Y1M10DT10H0M0S ... }'.

12.11.3 Within time intervals

A selection expression can be used to describe a time interval start or end.

EXAMPLE 1 '2018Y9ML1K1IN/P5D', with the expression contain selection describing a single date, represents the time interval "first Monday in September for a duration of 5 days".

EXAMPLE 2 '2018Y9ML{1,3}K1IN/P5D', with the expression contain selection describing a time interval set, represents the "first and third Monday in September, for a duration of 5 days". The expression is identical to the set expression '{ 2018Y9ML1K1IN/P5D, 2018Y9ML3K1IN/P5D }'.

13 Recurring time intervals with repeat rules

13.1 General

This clause extends [ISO 8601-1:2019](#), 5.6 on recurring time interval, by adding a rule part that defines the repeat pattern. The rule part is appended to the recurring time interval structure.

A representation for recurring time intervals with repeat rules allows the expression of a specific time interval along with rules for repeating occurrences of that time interval.

Specifically, this clause describes a repeat rule representation based on the mechanisms provided in CalConnect CC 18012, Clause 6, "Recurring time intervals with repeat rules", which is a superset but mostly interchangeable with syntax specified in RFC 5545.

This clause depends on the representations defined in [Clause 11](#) and [Clause 12](#).

13.2 Method of specification

A recurring time interval is represented with the following parts:

- optionally, a number of occurrences; if absent, the number of occurrences is unbounded;
- a time interval, as specified in [7.14](#) and in [ISO 8601-1:2019](#), 5.5.1;
- a repeat rule.

13.3 Specification of time interval

[ISO 8601-1:2019](#), 5.5.1 required the specified time interval to identify the first occurring time interval. This requirement is provisionally relaxed in this clause.

The interactions of a repeat rule are not easily predicable when the repeat rule includes the (optional) [selection-part] (see [13.4](#)). Therefore when the repeat rule includes the [selection-part], the time interval specified identifies only the start and end times of occurrences beyond the first and does not necessarily point to the start of the first occurrence.

13.4 Repeat rule

13.4.1 General

A repeat rule identifies a set of matching instants according to specification of a repeating cycle used together with selection rules.

repeat-rule = [eligibility-part][selection-part]

where

[eligibility-part] is the mandatory "eligibility part", represented as ["F"][eligible-time-intervals];

[selection-part] is the optional "selection part", represented as [selection], as specified in [Clause 12](#).

When selection rules from the selection part are applied to the specified eligible time intervals of the eligibility part, a set of matching time intervals is determined.

13.4.2 Eligibility part and eligible time intervals

The eligibility part determines the frequency of the repeating time intervals ("repeating intervals") and the gaps between eligible time intervals.

The designator ["F"] precedes the identification of the repeating intervals. Within each repeating interval, one sub-interval is distinguished, called an "eligible time interval". An eligible time interval is a time interval that contains instants that are eligible for output selection (the matching instants).

The representation of the eligibility part is given as follows.

eligibility-part = ["F"][eligible-time-intervals]

where [eligible-time-intervals] accepts one of the following time interval representations:

- one or more calendar years: [yearE];
- one or more calendar months: [monthE];
- one or more calendar weeks: [weekE];
- one or more calendar days: [dayE];
- one or more clock hours: [hourE];
- one or more clock minutes: [minE];
- one or more clock seconds: [secE].

NOTE The choice of designation symbol "F" reflects "frequency", as the value following it determines the frequency of the eligible time intervals, and also "for", where the expression "F2Y" could be said as "for every two years".

The value of [eligible-time-intervals] describes the following properties:

- a) The duration of each repeating interval is determined by the coefficient and time scale unit of the specified [eligible-time-intervals].
- b) The duration of each eligible time interval is one-unit of the chosen time scale component in which [eligible-time-intervals] is expressed.

EXAMPLE 1 If the value of [eligible-time-intervals] is '8Y', then the time scale component is year, the duration of each repeating time interval is 8 years, and each eligible interval is of a 1-year duration.

- c) Each eligible time interval begins x-1 units of the selected time scale component following the beginning of its repeating interval, where x is the coefficient of the unit.

EXAMPLE 2 If the value of [eligible-time-intervals] is '8Y', the eligible time interval is the 7th year within the 8-year repeating interval.

These properties together determine an exact set of eligible time intervals.

EXAMPLE 3 In the eligibility part, the expression 'F2Y' places the eligible time intervals as the second year of each 2-year repeating interval.

EXAMPLE 4 In the eligibility part, the expression 'F8D' places the eligible time intervals as the 8th day of each 8-day interval.

EXAMPLE 5 In the eligibility part, the expression 'F1Y' mean that the repeat intervals as well as the eligible time intervals are yearly.

13.4.3 Selection part and selection rules

The selection part provides a list of one or more selection rules, which specify conditions of matching one or more instants within a set of time intervals. Representations for possible selection rules are specified in [Clause 12](#).

When used with a set of eligible time intervals, the selection part identifies a set of instants within the eligible time intervals that match the selection rules. Each instant marks the beginning of an occurrence (a recurring interval).

13.5 Representations

A representation of a recurring time interval with repeat rules is as follows:

["R"][!][i]["/"][time-interval]["/"][repeat-rule]

where

- ["R"] is the recurring time interval designator;
- [i] is the number of recurrences (optional);
- [time-interval] is a valid time interval;
- [repeat-rule] is a repeat rule defined in [13.4](#).

The following examples all show a recurring interval of 12 occurrences, each of a 90-minute duration. The first occurrence begins September 29, 2015 at 2 pm. The second occurrence begins two weeks later, and so on.

EXAMPLE 1 'R12/20150929T140000/20150929T153000/F2W', interval in implicit form, specified by start and end. Since there is no selection part (see [13.4.1](#)), the repeat rule is simply 'F2W', which says that the intervals begin every two weeks.

EXAMPLE 2 'R12/20150929T140000/P1H30M0S/F2W', interval in implicit form, specified by start and duration.

EXAMPLE 3 'R12/P2H30M0S/20150929T153000/F2W', interval in implicit form, specified by duration and end.

13.6 Evaluation of a repeat rule

13.6.1 General

A repeat rule specifies a set of occurrences where each occurrence is a time interval.

The occurrences resulting from a repeat rule are calculated as follows:

- determine the eligible time intervals;
- apply all selection rules to the eligible time intervals; and
- obtain the resulting occurrences.

Methods to calculate duration in repeat rules are provided in [Annex D](#).

A detailed explanation of interactions between eligible time intervals and the selection part is provided in [Annex B](#).

Evaluation compatibility with RFC 5545 is discussed in [Annex C](#).

EXAMPLE 1 The expression 'R/2018-08-08/P1D/F1YL{3,8}M8DN', means "for every year, select the calendar months 3 (March) and 8 (August), calendar day 8, for an occurrence of 1-day duration". The resulting occurrences are therefore { '2018-08-08/2018-08-09', '2019-03-08/2019-03-09', '2019-08-08/2019-08-09', ... }.

EXAMPLE 2 The expression 'R/2018-08-01T10:20:00/PT10M/F1ML{1,10}DT10H20M0SN', means that "for every month, select the calendar days 1 and 10, time 10:20:00, for an occurrence of 10-minute duration". The resulting occurrences are therefore { '2018-08-01T10:20:00/2018-08-01T10:30:00', '2018-09-01T10:20:00/2018-09-01T10:30:00', ... }.

EXAMPLE 3 The expression 'R/2018-09-05/P1D/F1YL9M3K1IN', means "for every year, select the first Wednesday in September for an occurrence of 1-day duration". The resulting occurrences are therefore { '2018-09-05/2018-08-06', '2019-09-04/2019-09-05', '2020-09-02/2020-09-03', ... }.

13.6.2 Time scale unit precision

The resulting occurrences of a repeat rule shall have a time scale unit resolution equal to the lowest order time scale unit specified in the repeat rule.

EXAMPLE 1 In the expression 'R/2018Y1M/P1M/F3M', the lowest order time scale unit specified is month, hence the resolution is month precision. This expression resolves to the set { 2018-01/2018-02, 2018-04/2018-05 ... }.

EXAMPLE 2 In the expression 'R/2018Y1M1D/P1D/F3M', the lowest order time scale unit specified is day, hence the resolution is day precision. This expression resolves to the set {2018-01-01/2018-01-02, 2018-04-01/2018-04-02 ... }.

EXAMPLE 3 In the expression 'R/2018Y1M1DT0M/PT10M/F1M', the lowest order time scale unit specified is minute, hence the resolution is minute precision. This expression resolves to the set { 2018-01-01T00:00/2018-01-01T00:10, 2018-02-01T00:00/2018-02-01T00:10, ... }.

13.6.3 Inheritance of component values from time interval start

In a repeat rule with a selection part, the selection rules specified do not always fully account for all the time scale components provided in the initial start time, i.e. when there is no selection part, or if there is no one-to-one correspondence of time scale units used in the initial start time and those used in the selection part.

In this case, the selection rules shall inherit from time interval start the values of time scale units that are of lower order than those used in the eligibility part.

EXAMPLE 1 'R/20150104T083000/PT15M00S/F2YL1M1KT{8,9}H30M' expresses a recurring interval (the number of occurrences is unspecified) whose first occurrence is January 4, 2015, 8:30 to 8:45 AM, and subsequent occurrences, all of the same duration (15 minutes), are determined by the repeat cycle for which the following evaluation sequence is provided:

- the expression '2Y' indicates that the eligible time intervals have a repeating cycle of two years, and each eligible time interval is 1 year in length, the second year within its repeating interval;
- from this information together with the specification of the first occurrence, it is calculated that:
 - the first eligible time interval is the calendar year 2015 (the year during which the first occurrence takes place);
 - the first repeating interval is the two-year period comprising calendar years 2014 and 2015;
- the subsequent recurring intervals are then determined by the selection part:
 - the expression '1M' indicates that the matching occurrences are limited to January only;
 - the expression '1K' indicates that the matching occurrences are limited to Sundays only;
 - the expression '{8,9}H' indicates that the matching occurrences have clock hours 8 or 9;
 - the expression '30M' indicates that the matching occurrences have a clock minute value of 30, combined with specified clock hours, the starting times are determined to be 8:30 AM and 9:30 AM;
 - since the selection rules lacks specified values for clock seconds, in accordance with [13.6.3](#), they are obtained from the clock seconds value of the "time interval start" of '20150104T083000', hence the clock seconds selection rule is specified as value '00', the expression is therefore expanded to 'R/20150104T083000/PT15M00S/F2YL1M1KT{8,9}H30M0SN';
 - the recurrent occurrences therefore resolve to the rule "in the second year of every two year period beginning year 2014, for every Sunday in January at both 8:30:00 AM and 9:30:00 AM, create a 15 minutes occurrence."

EXAMPLE 2 Given the expression 'R/2018-08-01T01:02:03/PT5M/F1D', there is no selection part, however, the selection part is inferred from the time scale units of lower order than that of the eligibility part 'F1D', the "calendar day" unit. The specific time scale units specified in the time interval start and of lower order than "calendar day" are clock hour, clock minute and clock second. The inferred selection part directly takes values for those units from time interval start, resulting in the selection part 'LT1H2M3SN'. Therefore, the expression is expanded into 'R/2018-08-01T01:02:03/PT5M/F1DLT1H2M3SN'. The resulting occurrences are then {'2018-08-01T01:02:03/2018-08-01T01:07:03', '2018-08-02T01:02:03/2018-08-02T01:07:03', ... }.

EXAMPLE 3 Given the expression 'R/2018Y8M1DT1H/P1D/F2ML{1,3}D', the eligible time intervals are the second month of a 2-month period, selecting for the calendar days "1" and "3". Since the time scale units of lower order than calendar day ("D") do not have specified units, the clock hours from the initial start date are inherited in the selection rule, resulting in the selection part 'L{1,3}DT1H'. Therefore, the expression is expanded into 'R/2018Y8M1DT1H/P1D/F2ML{1,3}D L{1,3}DT1H'. The resulting occurrences are {'2018Y8M1DT1H/2018Y8M2DT1H', '2018Y8M3DT1H/2018Y8M4DT1H', '2018Y10M1DT1H/2018Y10M2DT1H', '2018Y10M3DT1H/2018Y10M4DT1H', ... }.

14 Date and time arithmetic

14.1 General

The modification of a duration with a scalar value, or with another duration, or the modification of a date/time by duration are described as "duration formulas".

Mechanisms provided here are based on CalConnect CC 18011, Clause 8, "Evaluation of date and time with duration".

The operational precedence of date and time arithmetic follows mathematical arithmetic, where:

- "+" represents addition;
- "-" represents subtraction;
- "x" represents multiplication; and
- parentheses "(" and ")" represent precedence grouping of operations.

Within a duration formula, a mixture of duration representations is allowed.

14.2 Addition and subtraction

Modification of a duration by the addition of another duration is described using the following representation.

$$\text{duration} = \text{durationA} + \text{durationB}$$

where

[duration] is the resolved duration;

[durationA] and [durationB] are durations to be summed.

EXAMPLE 1 'P3M2D + P1D' is a valid duration formula that uses composite duration.

EXAMPLE 2 'P3MP2D + P1D' is a valid duration formula that uses precedence duration.

The subtraction of duration is equivalent to the addition of a negative duration, of which direction is reversed on the time scale.

Hence, the two representations below are identical.

$$\text{duration} = \text{durationA} - \text{durationB}$$

$$\text{duration} = \text{durationA} + (-\text{durationB})$$

where

[duration] is the resolved duration;

[durationA] and [durationB] indicate that [durationA] is to be subtracted with [durationB].

In composite duration, addition of two durations results in a new duration where each time scale component is the sum of the corresponding time scale components of both durations. Note that the granularity of the resulting duration will be that of the finer granularity of the two durations.

$$[\text{compA1}]...[\text{compAn}] + [\text{compB1}]...[\text{compBn}] = [\text{value}(\text{"compA1"}) + \text{value}(\text{"compB1"})][\text{symbol}(\text{"compA1"})]...[\text{value}(\text{"compAn"}) + \text{value}(\text{"compBn"})][\text{symbol}(\text{"compAn"})]$$

where

[compAi] and [compBi] are the i-th durational units of [durationA] and [durationB], respectively;

[value(c)] is the scalar value of component [c];

[symbol(c)] is the designated symbol for the time scale component [c].

EXAMPLE 3 'PT40M60S + PT40M80S' results in 'PT80M140S'.

EXAMPLE 4 'P1Y10M3D + P2Y5M' results in 'P3Y15M3D'.

EXAMPLE 5 'PT40M60S - PT40M20S' results in 'PT40S'.

EXAMPLE 6 'P1Y10M3D - P2Y5MT10M' results in 'P3Y15M3DT-10M'.

EXAMPLE 7 'P3M2D - P1D' is equivalent to 'P3M2D + (-P1D)' and results in 'P3M1D'.

EXAMPLE 8 '-P3M4D - P5D' results in '-P3M9D'.

In precedence duration, addition of two durations results in a concatenation of durational units, without actual calculation of the time scale component values. This is to preserve the precedence order of the duration.

$$[\text{durA1}]...[\text{durAn}] + [\text{durB1}]...[\text{durBm}] = [\text{durA1}]...[\text{durAn}][\text{durB1}]...[\text{durBm}]$$

where [durAi] and [durBi] are the i-th durational units of [durationA] and [durationB], respectively.

EXAMPLE 9 'PT40MPT60S + PT40MPT80S' results in 'PT40MPT60SPT40MPT80S'.

EXAMPLE 10 'P1YP10MP3D + P2YP5M' results in 'P1YP10MP3DP2YP5M'.

EXAMPLE 11 'P1YP-10MP3D + P2YP-5M' results in 'P1YP-10MP3DP2YP-5M'.

As specified in [11.3.3](#), the external minus sign of a precedence duration can be absorbed into its durational units.

EXAMPLE 12 '-P3MP4D - P5D' results in 'P-3MP4DP-5D'.

EXAMPLE 13 'P3MP2D - P1D' results in 'P3MP2DP-1D'.

EXAMPLE 14 'P3D - P3M' results in 'P3DP-3M'.

EXAMPLE 15 '-P3M + P2Y' results in 'P-3MP2Y'.

14.3 Multiplication

In a duration formula, a duration may be multiplied by a scalar value, as in the following representation, resolving to a new duration.

$$\text{duration} = \text{coefficient} \times \text{durationA}$$

where

[duration] is the resolved duration;

[coefficient] is the scalar value to multiply by [durationA];

[durationA] is the duration to be modified.

The representation can also be shortened as:

$$\text{duration} = \text{coefficient durationA}$$

EXAMPLE 1 '3 × P3M2D' (or '3P3M2D') results in 'P9M6D'. In precedence duration, '3 × P3MP2D' results in 'P9MP6D'.

EXAMPLE 2 '0.5 × P3M2D' results in 'P1.5M1D'. In precedence duration, '0.5 × P3MP2D' results in 'P1.5MP1D'.

The method of calculation for fractional duration values is given in [D.4.5](#).

14.4 Date time modified by duration

A date and time representation may be modified by a duration as in the following representation.

$$\text{date} = \text{dateA} + \text{durationB}$$

where

[date] is the resolved date;

[dateA] is a non-duration date and time expression to be modified;

[durationB] is a duration expression.

EXAMPLE 1 '2018Y9M10D + P1D' is a valid date time formula resolving to '2018Y9M11D'.

EXAMPLE 2 '2018Y9M10DT9H10M + P2MT50S' resolves to '2018Y11M10DT9H10M50S'.

If [durationA] is in precedence representation, the evaluation order is from the leftmost durational unit to the rightmost.

EXAMPLE 3 The expression '2018Y9M10DT9H10M + P2MPT50S' is identical to the effect of '(2018Y9M10DT9H10M + P2M) + PT50S', where the parentheses indicate precedence order.

EXAMPLE 4 The expression '2020Y2M29D + P2DP2Y' is identical to the effect of '(2020Y2M29D + P2D) + P2Y', where the parentheses indicate precedence order.

Methods to calculate date time formulas are given in [Annex D](#), including the calculation for:

- time intervals when given duration as a start or end;
- boundaries of grouped time scale units ([Clause 5](#));
- conversion between of UTC of day and local time of day via time shifts; and
- occurrences of recurring time intervals.

15 Profiles

15.1 General

The [ISO 8601](#) series includes many features and in many cases allows several different formats to represent a single feature or multiple interpretations for a single format. Vendors implementing the [ISO 8601](#) series may implement only a subset of its features, or different representations of a given feature, causing interoperability issues with other implementations.

An [ISO 8601](#) profile is a specification of how the [ISO 8601](#) series is to be used for a particular context (application, discipline or community), specifying the necessary features and representations to implement and providing interpretations applicable to the particular context.

This document provides one such profile in [Annex A](#).

NOTE This document supports the creation of a registration agency for [ISO 8601](#) profiles, which registers profiles upon request, helps assure uniqueness of profile names and facilitates the distribution of such profiles.

15.2 Requirements

An [ISO 8601](#) profile should at least contain:

- minimum features of the [ISO 8601](#) series that shall be supported to conform to the profile;

and may optionally contain:

- features of the [ISO 8601](#) series that are not relevant and need not be supported;
- in cases where there are multiple methods specified in the [ISO 8601](#) series to support a particular function, selection of specific methods;
- in cases where there are different interpretations of a particular function, the profile's interpretation or clarification;
- conformance levels, if any (requirements for each conformance level shall be clearly stated).

Annex A (informative)

Profile: Extended Date/Time Format

A.1 General

The Extended Date/Time Format (EDTF) specification (Library of Congress EDTF) was created by the Library of Congress with the participation and support of the bibliographic community as well as communities with related interests.

A.2 Compliance

This profile specifies three conformance levels: level 0, level 1 and level 2.

- level 0 specifies features of [ISO 8601-1](#);
- levels 1 and 2 specify features described in this document.

An implementation of this specification must support all of the features listed for level 0. The vendor must state one of the following levels of support:

- Level 0 is supported.
- Level 0 is supported, and in addition the following features of levels 1 and 2 are supported (list features).
- Level 1 is supported.
- Level 1 is supported, and in addition the following features of level 2 are supported (list features).
- Level 2 is supported.

Two communication parties that agree to operate according to this profile must suppress, during their communication, any features of [ISO 8601-1](#) that are not included in level 0.

A.3 Extended format

This profile requires implicit form "extended format" (separators between time scale components) for the representation of dates and times. Implicit form basic format and explicit form representations are not used.

A.4 Level 0

A.4.1 General

Level 0 of this profile requires support for the following features.

A.4.2 Date

This profile adopts the following representations for a calendar date.

- complete representation as specified in [ISO 8601-1:2019](#), 5.2.2.1, b), "Extended format":

[year]["-"][month]["-"][day]

EXAMPLE 1 '1985-04-12' refers to the calendar date 1985 April 12th.

- reduced precision, for year and month, as specified in [ISO 8601-1:2019](#), 5.2.2.2, a), "Extended format":

[year]["-"][month]

EXAMPLE 2 '1985-04' refers to the calendar date 1985 April with month precision.

- reduced precision, for year, as specified in [ISO 8601-1:2019](#), 5.2.2.2, b):

[year]

EXAMPLE 3 '1985' refers to the calendar date of year 1985 with year precision.

A.4.3 Date and Time

This profile adopts the following complete representations for date and time of day specified in [ISO 8601-1:2019](#), 5.4.2.1, b), on the extended format of the complete representation of a calendar date.

- a) Date and time

[dateI]["T"][timeI]

EXAMPLE 1 '1985-04-12T23:20:30' refers to the date 1985 April 12th 23:20:30 in local time of day.

- b) Date and time with no time shift from UTC

[dateI]["T"][timeI]["Z"]

EXAMPLE 2 '1985-04-12T23:20:30Z' refers to the date 1985 April 12th 23:20:30 in UTC of day.

- c) Date and time with hour and minutes time shift

[dateI]["T"][timeI][shiftX]

EXAMPLE 3 '1985-04-12T23:20:30+04:30' refers to the date 1985 April 12th, time of day 23:20:30 with time shift of 4 hours and 30 minutes ahead of UTC.

- d) Date and time with hour time shift

[dateI]["T"][timeI][shiftH]

EXAMPLE 4 '1985-04-12T23:20:30+04' refers to the date 1985 April 12th time of day 23:20:30 in a time of day with time shift of 4 hours ahead of UTC.

A.4.4 Time interval

This profile adopts representations of a time interval as described in [ISO 8601-1](#), where both the start and end are dates (no durations and no times included).

EXAMPLE 1 '1964/2008' is a time interval with calendar year precision, beginning sometime in 1964 and ending sometime in 2008.

EXAMPLE 2 '2004-06/2006-08' is a time interval with calendar month precision, beginning sometime in June 2004 and ending sometime in August 2006.

EXAMPLE 3 '2004-02-01/2005-02-08' is a time interval with calendar day precision, beginning sometime on February 1, 2004 and ending sometime on February 8, 2005.

EXAMPLE 4 '2004-02-01/2005-02' is a time interval beginning sometime on February 1, 2004 and ending sometime in February 2005. Note that the start endpoint precision (year) is different than the end endpoint (month) and therefore the precision of the time interval at large is undefined.

EXAMPLE 5 '2004-02-01/2005' is a time interval beginning sometime on February 1, 2004 and ending sometime in 2005. The start endpoint has calendar day precision and the end endpoint has calendar year precision. Similar to the previous example, the precision of the time interval at large is undefined.

EXAMPLE 6 '2005/2006-02' is a time interval beginning sometime in 2005 and ending sometime in February 2006.

A.5 Level 1

A.5.1 General

Level 1 of this profile requires support for level 0 as well as the following features.

A.5.2 Extended Year

- [4.7.2](#) letter-prefixed calendar year

EXAMPLE 1 'Y1700000002'.

- [4.4.1.2](#), a) negative calendar year

EXAMPLE 2 '-1985'.

NOTE [ISO 8601-1](#) does not permit negative values for four-digit years.

A.5.3 Seasons

- [4.8.3](#), a) sub-year groupings — groupings represented as months — basic seasons in the month time scale component

EXAMPLE '2001-21' (Spring).

A.5.4 Qualification

- [8.4.1](#), a) qualification of a complete date (implicit format)

EXAMPLE 1 '1985-04-12?'.

- [8.4.2](#), a) qualification of a specific calendar year and month

EXAMPLE 2 '1985-04?'.

- [8.4.2](#), b) qualification of a specific calendar year

EXAMPLE 3 '1985~'.

A.5.5 Unspecified digits

- [9.2.1.1](#), a) calendar year and calendar month specified, calendar day unspecified

EXAMPLE 1 '1985-04-XX'.

- [9.2.1.1](#), b) unspecified-time-scale-component-right-complete-year calendar year specified, calendar month and calendar day unspecified

EXAMPLE 2 '1985-XX-XX'.

- [9.2.1.2](#), a) reduced precision, calendar year specified, calendar month unspecified

EXAMPLE 3 '2004-XX'.

- [9.2.1.2](#), c), 1) and [9.2.1.2](#), c), 2) reduced precision, one or two rightmost calendar year digits unspecified.

EXAMPLE 4 '201X', '20XX'.

A.5.6 Extended interval

- [10.2](#), a) through d) unknown or open start or end time intervals

EXAMPLE 1 '1985-04-12/..'

EXAMPLE 2 '..-/1985-04-12'

EXAMPLE 3 '1986-04/'

EXAMPLE 4 '/1985'

- [10.3.2](#) qualification of dates in time intervals — complete qualification

EXAMPLE 5 '1984?/2004%'.

EXAMPLE 6 '1984-01-02~/2004-06-04'.

EXAMPLE 7 '1984~/2004-06'.

- [10.2](#) unknown or open start or end time intervals with qualification

EXAMPLE 8 '..-/1985-04-12?'.

EXAMPLE 9 '1985-04-12~/.'

A.6 Level 2

A.6.1 General

Level 2 of this profile requires support for level 1 as well as the following features.

A.6.2 Exponential calendar year

As described in [4.7.3](#).

EXAMPLE 'Y-17E7', the calendar year $-17 \times 10^7 = -170000000$.

A.6.3 Significant digits (as applied to year component)

As described in [4.7.4](#).

EXAMPLE 1 '1950S2', some year between 1900 and 1999, estimated to be 1950.

EXAMPLE 2 'Y171010000S3', some year between 171010000 and 171010999, estimated to be 171010000.

EXAMPLE 3 'Y3388E2S3', some year between 338000 and 338999, estimated to be 338800.

A.6.4 Seasons

[4.8.3](#), b) sub-year groupings — groupings represented as months — hemispheric seasons and typical divisions in the month time scale component

EXAMPLE '2001-34', second quarter of 2001.

A.6.5 Set representation

- [6.1](#) set of date and time expressions

EXAMPLE 1 '{1960, 1961-12}' calendar year 1960 and the month December of 1961.

- [6.2](#) single element amongst set

EXAMPLE 2 '[1667, 1760-12]' calendar year 1667 or the month December of 1760.

- [6.3](#) range element expansion

- 1) [6.3](#), a) on or before a date and time expression, inclusive

EXAMPLE 3 '..1984'.

- 2) [6.3](#), b) on or after a date and time expression, inclusive

EXAMPLE 4 '1984..'

- 3) [6.3](#), c) all elements between two date and time expressions, inclusive

EXAMPLE 5 '1670..1673'.

- 4) [6.3](#), d) any combination of [6.3](#), a), [6.3](#), b) and [6.3](#), c)

EXAMPLE 6 '..1983-12-31,1984-10-10..1984-11-01,1984-11-05..'

A.6.6 Qualification

- [8.4.4](#) qualification of a group of time scale components from the right

EXAMPLE 1 '2004-06~-11' (qualification applies to month and year).

EXAMPLE 2 '2004?-06-11' (qualification applies to year only).

- [8.4.5](#) qualification of individual time scale components

EXAMPLE 3 '?2004-06-~11', calendar year uncertain; calendar month known; calendar day approximate.

EXAMPLE 4 '2004-%06-11', calendar month uncertain and approximate; calendar year and calendar day known.

- [10.3.3](#) partial qualification of dates in time intervals

EXAMPLE 5 '2004-06-~01/2004-06-~20'.

- [10.6](#) before and after with qualified time scale components

EXAMPLE 6 '..2004-06-01/~2004-06-20', beginning before or on 1 June 2004 and ending approximately the 20th.

EXAMPLE 7 '2004-06-01~/2004-06-20..', beginning approximately 1 June 2004 and ending the 20th or later.

A.6.7 Unspecified digits

- [9.2.2](#) unspecified digit anywhere in time scale component

EXAMPLE 1 '156X-12-25', December 25 sometime during the 1560s.

EXAMPLE 2 'XXXX-12-XX', some day in December in some year.

EXAMPLE 3 '1XXX-XX', some month during the 1000s.

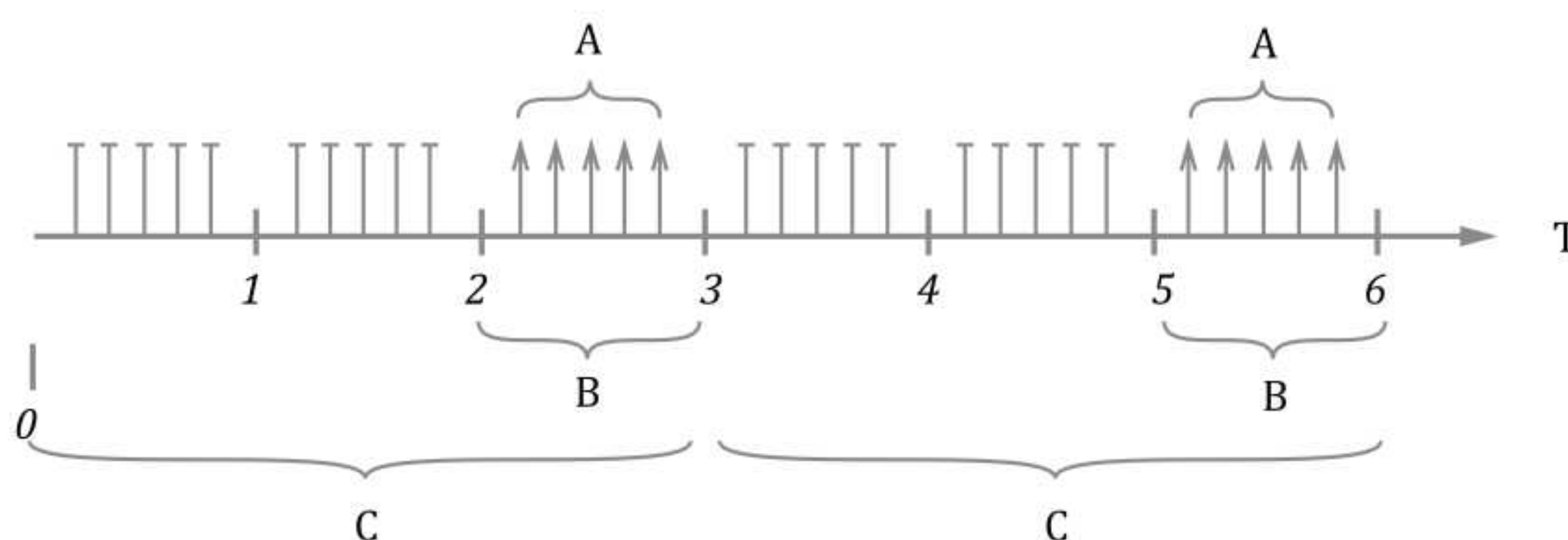
EXAMPLE 4 '1XXX-12', some December during the 1000s.

Annex B (informative)

Interactions between eligible time intervals with the selection part

B.1 General

The interaction between eligible time intervals and selection rules specified within a repeating rule give rise to interesting properties that users should be aware of.



Key

- T time in years
- A matching instants
- B eligible time interval
- C repeating cycle

NOTE The rule F3YL{2,4,6,8,10}M1DN: repeat every 3 years, in the last year, select months 2, 4, 6, 8, 10, the 1st day.

Figure B.1 — Resulting occurrences of the rule F3YL{2,4,6,8,10}M1DN

[Figure B.1](#) demonstrates that the repeating cycle denotes how often the eligible time intervals should be evaluated. Within the eligible time intervals, the selection rules are applied. It is the overlap between the selection rules and eligible time intervals that produce the resulting occurrences.

B.2 Special case when the repeating cycle uses value 1

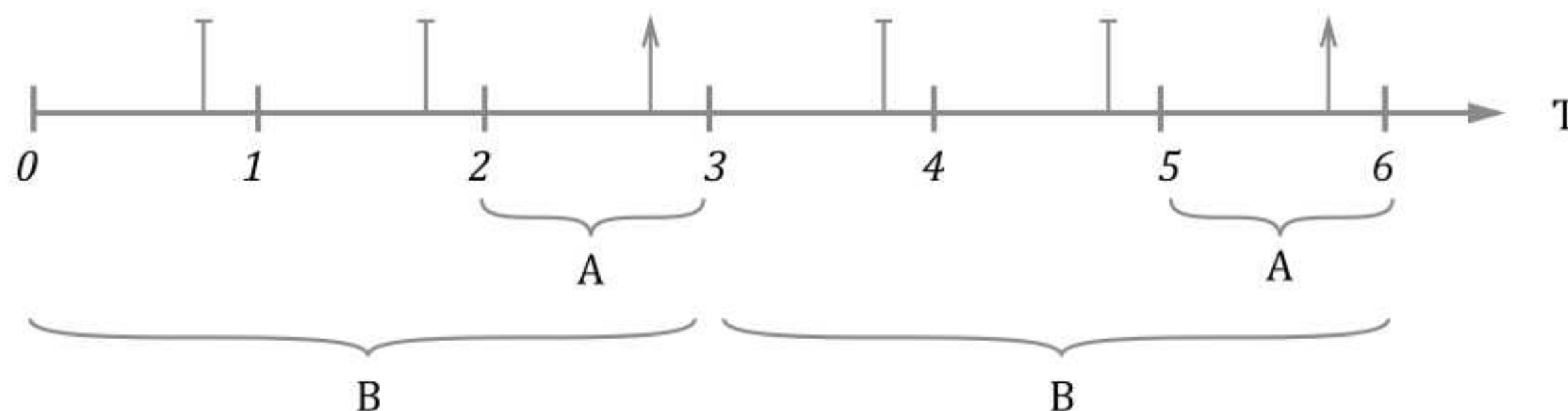
When the repeating cycle is defined with a value 1 for any time unit (e.g. calendar year, calendar month, calendar day, calendar hour, etc.), the effect on the resulting occurrences are identical — the repeating cycle fully covers all instants of the time scale. Therefore, the resulting occurrences are fully described by the selection rules that apply.

B.3 Orders of the repeating cycle and selection rules

B.3.1 Repeating cycle of higher order than selection rules

It is common in natural expressions and in calendar implementations that the repeating cycle uses a time scale unit of a higher order than that of the selection rules. The resulting occurrences are generally as expected by the creator of these rules.

EXAMPLE [Figure B.2](#) provides such a case; where the resulting occurrences happen once every three years, matching a single date of September 10th.



Key

T time in years

A eligible time interval

B repeating cycle

↑ matching instants

⊥ matching but outside of eligible time intervals

NOTE The rule F3YL9M10DN: repeat every 3 years, in the last year, select month 9, the 10th day.

Figure B.2 — When the repeating cycle is of a higher order than the selection part

B.3.2 Repeating cycle of same order with selection rules

When a time scale unit of the same order is used for both the repeating cycle and the selection rules, the following properties arise:

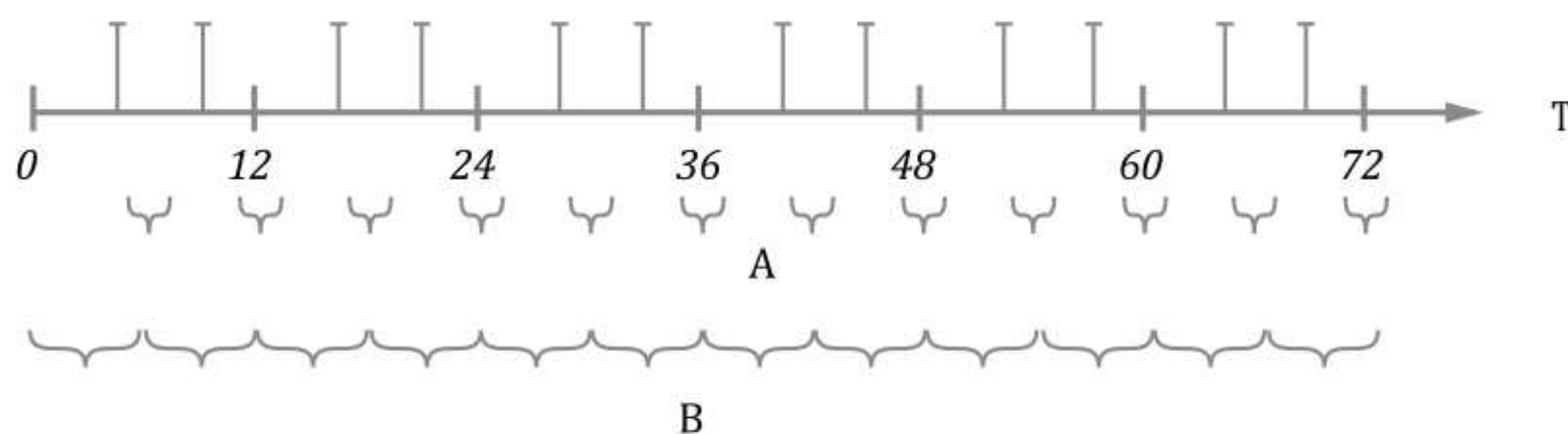
- The effect of [B.2](#) applies.

EXAMPLE 1 A repeating rule of 1 month repeating cycle, with selection rules that are of the highest order of "month", has the same effect as the repeating cycle of 1 calendar year because every calendar month in the calendar year will be evaluated.

- A repeating rule with an n time unit repeating cycle, matched with selection rules of the same time unit, will provide occurrences that depend on the start instant of the repeating cycle.

EXAMPLE 2 A repeating cycle starting in April every 6 months will only match a monthly selection rule that contains April or October.

EXAMPLE 3 [Figure B.3](#) demonstrates an instance of the second case where the repeating cycle does not overlap with eligible time intervals, resulting in no occurrences.



Key

- T time in months
- A eligible time intervals
- B repeating cycles
- ┤ matching but outside of eligible time intervals

NOTE The rule F6ML{5,9}M1DN: repeat every 6 months, in the last month, select months 5 and 9, the 1st day.

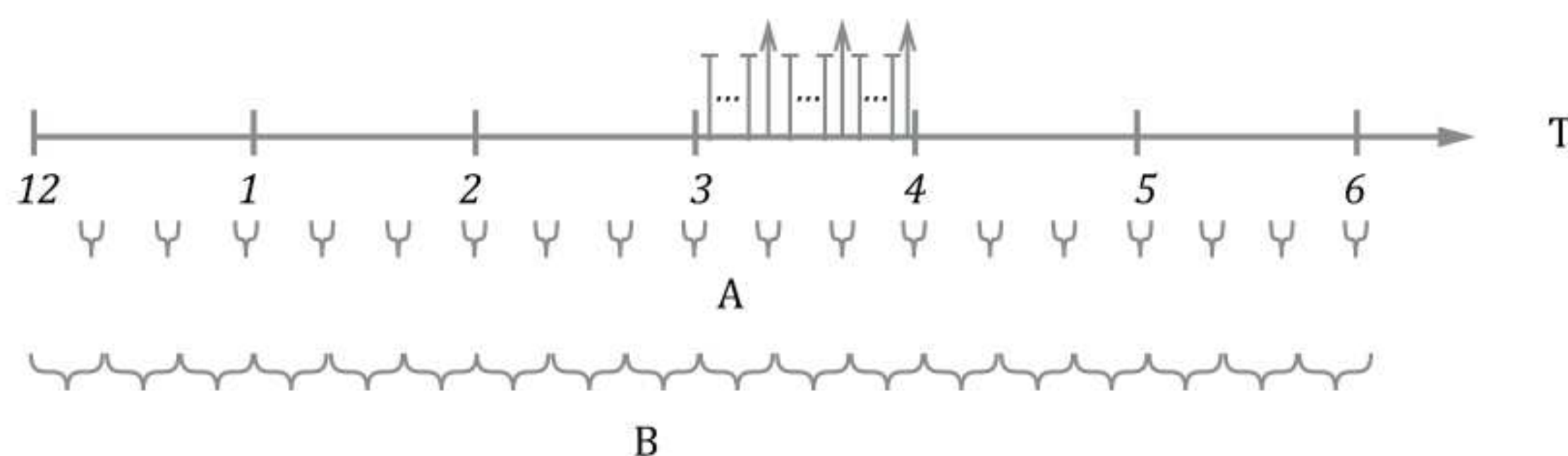
Figure B.3 — When the repeating cycle is of the same order as the selection part and mismatches

B.3.3 Repeating cycle of lower order than selection rules

When a time scale unit of a lower order is used for the repeating cycle than that of the selection rules, the following should be of note:

- The effect of [B.2](#) applies.
- A repeating rule with an n time unit repeating cycle, matched with selection rules of a lower order time unit, will provide occurrences that depend on the start instant of the repeating cycle.

EXAMPLE [Figure B.4](#) demonstrates this interaction of the second case, where the repeating cycle is of day order and a selection rule of calendar month order. Notice that there are no matches outside calendar month 3 due to the application of the selection rule.



Key

- T time in months
- A eligible time intervals
- B repeating cycles
- ↑ matching instants
- ┤ matching but outside of eligible time intervals

NOTE The rule F10DL3MN: repeat every 10 days, in the last day, select month 3.

Figure B.4 — When the repeating cycle is of a lower order than the selection part

Annex C (informative)

Compatibility considerations of repeat rules with IETF RFC 5545 recurrences

C.1 Evaluation of repeat rules

In this document, the evaluation of repeat rules (see [13.4](#)) rely on explicit specification of selection rules (see [Clause 12](#)) and the direct inheritance of time scale component information from the initial start date.

C.2 Inheritance of time scale component information

In the evaluation of repeat rules within this document as well as in RFC 5545, a number of time scale components are directly inherited from the initial start date.

The following parameters, when unspecified, have values directly inherited from that of initial start date:

- 'BYSECOND' for clock seconds;
- 'BYMINUTE' for clock minutes;
- 'BYHOUR' for clock hours.

In terms of RFC 5545 specifically:

- when the 'FREQ' parameter is set to 'MINUTELY', but without a 'BYSECOND' parameter, the 'BYSECOND' selection is directly inherited from the clock seconds value from the initial start date;
- when the 'FREQ' parameter is set to 'HOURLY', but without a 'BYMINUTE' parameter, the 'BYMINUTE' selection is directly inherited from the clock minutes value from the initial start date, similarly for any time scale unit of lower order than clock minute (i.e. 'BYSECOND');
- when the 'FREQ' parameter is set to 'DAILY', but without a 'BYHOUR' parameter, the 'BYHOUR' selection is directly inherited from the clock hours value from the initial start date, similarly for any time scale unit of lower order than clock minute (i.e. 'BYMINUTE' and 'BYSECOND').

C.3 Implicit selection rules of RFC 5545

In RFC 5545, however, the evaluation of certain repeat rules also relies on implicit selection rules inherited indirectly from the initial start date.

Specifically,

- when the 'FREQ' parameter is set to 'WEEKLY', but without a 'BYDAY' parameter, the 'BYDAY' selection is inherited from the calendar day of week value from the initial start date (note that the calendar day of week value is not directly specified in the initial start date, but it has to be inferred);
- when the 'FREQ' parameter is set to 'MONTHLY', but without both 'BYMONTHDAY' and 'BYDAY' parameters, the 'BYMONTHDAY' selection is inherited from the calendar month of year value from the initial start date;

- when the 'FREQ' parameter is set to 'YEARLY' but without a 'BYYEARDAY' parameter,
 - if no 'BYMONTH' or 'BYWEEKNO' parameter is set:
 - if the 'BYMONTHDAY' parameter is provided, then the 'BYMONTH' selection is inherited from the calendar month of year value from the initial start date;
 - if the 'BYDAY' parameter is not set, then the 'BYMONTH' selection is inherited from the calendar month of year value from the initial start date;
 - if no 'BYMONTHDAY', 'BYWEEKNO' or 'BYDAY' parameter is set, the 'BYMONTHDAY' selection is inherited from the calendar day of month of the initial start date;
 - if there is a 'BYWEEKNO' parameter set but no 'BYMONTHDAY' or 'BYDAY', the 'BYDAY' selection is inherited from the calendar day of week of the initial start date.

EXAMPLE In evaluating a simplified example expression from RFC 5545, with 'DTSTART' set to '19970902T090000' and 'RRULE' set to 'FREQ=WEEKLY;INTERVAL=2', will result in the instance series of "1997 September 2, 16, 30; October 14...". This resulting instance series relies on an implicit understanding that 'FREQ=WEEKLY' always requires selection of the 'BYDAY' parameter, which is not specified in the original selection rule. In this case, 'BYDAY' is implicitly set to Tuesdays as originally obtained from the 'DTSTART' value being a Tuesday.

C.4 Achieving equivalent selection rules in this document

Using mechanisms described in this document, implicit selection rules are not allowed. In order to convert a RFC 5545 recurrence rule into a [ISO 8601-2](#) repeat rule, the implicit selection rules based on indirect inheritance must be made into explicit selection rules.

EXAMPLE Following the example in [C.3](#), the value of Tuesday is considered to be indirectly inferred from the initial start date since it is not explicitly specified. To achieve the same effect using mechanisms of this document, the 'BYDAY' selection rule in RFC 5545 is supposed to be explicitly set as a selection rule, such as in 'L1KN'.

Annex D (informative)

Evaluation of date time formulas and duration considerations

D.1 General

This Annex provides methods to evaluate date time formulas and their components in a consistent manner.

D.2 Calculation of exact duration

D.2.1 General

This clause describes methods to resolve exact duration in face of ambiguities.

D.2.2 Context-dependent duration

The exact duration of some time scale units can only be known when placed on the actual time scale.

EXAMPLE 1 'P1Y' duration could be 365 or 366 calendar days depending on whether the year is a leap year.

EXAMPLE 2 'P3M' duration could be 90, 91 or 92 days depending on the actual calendar months and calendar year.

EXAMPLE 3 'PT1M' duration is normally 60 seconds, but when the minute is the last minute of the year, it could be 59, 60 or 61 seconds long depending on whether a leap second is applied.

EXAMPLE 4 'P1Y2M15DT12H30M', duration of 1 year, 2 months, 15 days, 12 hours and 30 minutes. The actual duration of the involved calendar months can only be known when the duration is placed on an actual start date.

D.2.3 Speculative duration

The exact duration of certain time scale unit can only be known after an event occurs.

There is no universal rule to calculate leap seconds in advance, since the decision to insert a leap second is driven by a number of dynamic factors and only known when announced by the Bureau international des poids et mesures (BIPM).

Specifically, the exact duration of the last clock minute of a calendar year can only be known after it is announced by the BIPM, whether a leap second has or will occur (in UTC). This event is described in this document to occur at the "leap second horizon date" for a calendar year.

The exact duration of the last clock minute in a calendar year is one of the following durations:

- when there is a positive leap second, 61 clock seconds;
- when there is no leap second, 60 clock seconds; and
- when there is a negative leap second, 59 clock seconds.

For the accuracy of general purposes, it is acceptable to assume the last clock minute always contains 60 clock seconds, just like any other clock minute.

For cases where durational accuracy at the clock second level is of importance, the following steps shall be taken:

- when calculating exact duration prior to the leap second horizon date of a specific calendar year involved, the last clock minute is assumed to have no leap second, such that '59' is always the last second of the year;
- when calculating exact duration after the leap second horizon date of a specific calendar year involved, the last clock minute shall reflect the actual duration.

EXAMPLE 1 'P3000Y12M31DT23H59M' is considered to have a duration of 60 clock seconds, as the calendar year 3000's leap second horizon date has not yet occurred.

EXAMPLE 2 'P2016Y12M31DT23H59M' is considered to have a duration of 61 clock seconds, since the leap second horizon date has already occurred, and it is known that a positive leap second assigned to the calendar year 2016.

D.3 Prerequisite mechanisms for date and time calculations

D.3.1 Carry-over of overflow in time scale components

An "overflow" of a date and time expression is defined as assigning a value exceeding the maximum value accepted by the time scale component.

EXAMPLE 1 An increase of 'P1M' (duration) to '2018Y12M' (date) will result in the expression '2018Y13M', where the month component is overflowed with value '13'.

An overflow is considered resolved once the overflowed time scale unit has transferred its excess to the immediate higher order time scale component.

EXAMPLE 2 The overflowed expression '2018Y13M' can be resolved to '2019Y1M'.

An overflow can cause multiple carry-overs when the overflow not only causes the immediate higher order time scale component to overflow, but also subsequent higher order components.

EXAMPLE 3 The overflowed expression '2018Y12M366D' can be resolved to '2018Y24M1D' (which still contains an overflow), which can be resolved to '2019Y12M1D' (where there is no more overflow).

D.3.2 Truncation at time scale component boundaries

A modification made to a higher order time scale component can cause the value of a lower order time scale component to go beyond its value boundaries. This condition is described as "out of bounds" of a date and time expression.

This situation occurs when a modification is made to the values of certain time scale components, because a lower order component always has its maximum value bound by its immediate higher order component:

- clock minute (clock second value of '60' is only valid for a leap second);
- calendar month (calendar day values '29', '30', '31' are invalid for some months); and
- calendar year (ordinal day value of '366' and calendar month/day combination of February 29th is only valid for a leap year).

EXAMPLE 1 An increase of 'P1M' (duration) to '2018Y1M31D' (date) will result in the expression '2018Y2M31D', which is an invalid date representation as the calendar day value of '31' has exceeded the boundary of its immediate higher order component, the month '2' (February).

EXAMPLE 2 An increase of 'P1Y' (duration) to '2016Y12M31DT23H59M60S' (date) will result in the expression '2017Y12M31DT23H59M60S', which is an invalid date representation as the clock second value of '60' has exceeded the boundary of its immediate higher order component, the clock minute with its last second at '59'.

An out of bounds situation is considered resolved once the excess value is truncated at the violating (lower order) time scale component.

EXAMPLE 3 The out of bounds expression '2019Y2M29D' can be truncated to '2019Y2M28D'.

EXAMPLE 4 The out of bounds expression '2020Y3660' can be truncated to '2020Y3650'.

D.4 Evaluation of date and time with duration

D.4.1 Simple duration

A simple duration is a duration expression with only a single time scale component.

Evaluation of a date time formula that contains a simple duration is to directly apply the durational unit to the corresponding time scale component of the date and time representation and applying carry-over.

The steps are as follows:

- Take the value of the time scale component in the simple duration, directly modify the value of the corresponding time scale component in the date and time representation.
- In the resulting date and time representation, starting from the lowest order overflowed time scale component, perform carry-over until there is no more overflow in the representation.
- If any time scale components have become invalid, apply truncation to them.
- The resulting date and time representation is complete.

EXAMPLE Calculation of '2022Y2M28D + P3D':

- Direct modification of date and time representation: '2022Y2M31D'.
- Apply carry-over to overflowed components: '2022Y3M02D'.

D.4.2 Composite duration

The method of evaluating of a date time formula with a composite duration ([11.3.2](#)) is to directly apply the durational units to the date and time representation without truncation and only apply carry-over at the end.

The steps are as follows:

- For all values of time scale components in the composite duration, directly modify the values of the corresponding time scale components in the date and time representation.
- In the resulting date and time representation, starting from the lowest order overflowed time scale component, perform carry-over on all overflowed time scale components until there is no more overflow in the representation.
- If any time scale components have become invalid, apply truncation to them.
- The resulting date and time representation is complete.

EXAMPLE 1 Calculation of '2022Y2M30D + P1Y3M2D':

- Direct modification of date and time representation: '2023Y5M32D'.
- Apply carry-over to overflowed components: '2023Y6M31D'.
- The resulting representation is '2023Y6M31D'.

EXAMPLE 2 Calculation of '2020Y2M29D + P2Y2M2D':

- Direct modification of date and time representation: '2022Y4M31D'.

- Apply carry-over to overflowed components: '2022Y5M1D'.
- The resulting representation is '2022Y5M1D'.

D.4.3 Precedence duration

The method of evaluating of a date time formula with a precedence duration ([11.3.3](#)) is to apply the durational units one by one to the date and time representation with truncation and carry-over at every step.

The steps are as follows:

- Starting at the left side of the precedence duration expression, process every durational unit one by one as follows:
 - Use the durational unit (which only contains one time scale component), to modify the corresponding time scale component (the "target component") in the date and time representation.
 - If the target component has overflowed in the previous step, perform carry-over (to higher order time scale components) until there is no more overflow in the representation.
 - If any time scale components of lower order than the target component have become invalid, apply truncation to them.
- Once all durational units have been processed, the resulting date and time representation is complete.

EXAMPLE 1 Calculation of '2022Y2M29D + P1YP3MP2D':

- Process the first durational unit: '(2022Y2M29D + P1Y) + P3MP2D'.
- Apply the first durational unit to the date and time representation: '2023Y2M29D + P3MP2D'.
- No carry-over is necessary since there is no overflow.
- Since '29D' is of lower order than the target component (calendar year), the representation is truncated: '2023Y2M28D + P3MP2D'.
- Process the second durational unit: '(2023Y2M28D + P3M) + P2D'.
- Apply the second durational unit to the date and time representation: '2023Y5M28D + P2D'.
- No carry-over is necessary since there is no overflow.
- No component of lower order than the target component is invalid, hence truncation is not necessary.
- Process the third durational unit: '2023Y5M28D + P2D'.
- Apply the third durational unit to the date and time representation: '2023Y5M31D'.
- No carry-over is necessary since there is no overflow.
- No component of lower order than the target component is invalid, hence truncation is not necessary.
- The resulting date and time representation is '2023Y5M31D'.

EXAMPLE 2 Calculation of '2022Y2M29D + P2DP3MP1Y':

- Process the first durational unit: '(2022Y2M29D + P2D) + P3MP1Y'.
- Apply the first durational unit to the date and time representation: '2022Y2M31D + P3MP1Y'.
- Carry-over is necessary since there is the target component has overflowed: '2022Y3M2D + P3MP1Y'.
- No component of lower order than the target component is invalid, hence truncation is not necessary.
- Process the second durational unit: '(2022Y3M2D + P3M) + P1Y'.

- Apply the second durational unit to the date and time representation: '2022Y6M2D + P1Y'.
- No carry-over is necessary since there is no overflow.
- No component of lower order than the target component is invalid, hence truncation is not necessary.
- Process the third durational unit: '2022Y6M2D + P1Y'.
- Apply the third durational unit to the date and time representation: '2023Y6M2D'.
- No carry-over is necessary since there is no overflow.
- No component of lower order than the target component is invalid, hence truncation is not necessary.
- The resulting date and time representation is '2023Y6M2D'.

EXAMPLE 3 Calculation of '2020Y2M29D + P2YP2MP2D':

- Process the first durational unit: '(2020Y2M29D + P2Y) + P2MP2D'.
- Apply the first durational unit to the date and time representation: '2022Y2M29D + P2MP2D'.
- No carry-over is necessary since there is no overflow.
- Truncation of lower order components is necessary since year 2022 does not have February 29th. The day has truncated: '2022Y2M28D + P2MP2D'.
- Process the second durational unit: '(2022Y2M28D + P2M) + P2D'.
- Apply the second durational unit to the date and time representation: '2022Y4M28D + P2D'.
- No carry-over is necessary since there is no overflow.
- No component of lower order than the target component is invalid, hence truncation is not necessary.
- Process the third durational unit: '2022Y4M28D + P2D'.
- Apply the third durational unit to the date and time representation: '2022Y4M30D'.
- No carry-over is necessary since there is no overflow.
- No component of lower order than the target component is invalid, hence truncation is not necessary.
- The resulting date and time representation is '2022Y4M30D'.

EXAMPLE 4 Calculation of '2020Y2M29D + P2DP2MP2Y':

- Process the first durational unit: '(2020Y2M29D + P2D) + P2MP2Y'.
- Apply the first durational unit to the date and time representation: '2020Y2M31D + P2MP2Y'.
- Carry-over is necessary since there is overflow at the day unit: '2020Y3M2D + P2MP2Y'.
- No component of lower order than the target component is invalid, hence truncation is not necessary.
- Process the second durational unit: '(2020Y3M2D + P2M) + P2Y'.
- Apply the second durational unit to the date and time representation: '2020Y5M2D + P2Y'.
- No carry-over is necessary since there is no overflow.
- No component of lower order than the target component is invalid, hence truncation is not necessary.
- Process the third durational unit: '2020Y5M2D + P2Y'.
- Apply the third durational unit to the date and time representation: '2022Y5M2D'.
- No carry-over is necessary since there is no overflow.

- No component of lower order than the target component is invalid, hence truncation is not necessary.
- The resulting date and time representation is '2022Y5M2D'.

D.4.4 Involving fractional duration

When fractional duration ([11.4](#)) is applied to context-dependent durational units ([D.2.2](#)), it is necessary to resolve the fractional duration in context.

EXAMPLE 1 The expression 'P0.5M' ("half a month") is ambiguous because the exact duration of a calendar month depends on its context, and that the context for which 'P0.5M' is anchored to is unclear.

The duration of a fractional, context-dependent and durational unit can be obtained by first calculating a single unit of the durational unit in context, such that the exact duration for the durational unit in whole (e.g. 'P1M') becomes known. Once the exact duration of a single unit is known, the duration of the fractional unit can be calculated.

$$\text{date} + \text{fracDur} = \text{duration}(\text{date} + \text{unit}(\text{fracDur}), \text{date}) \times \text{value}(\text{fracDur}) + \text{date}$$

where

- 'unit(fracDur)' is the value of a single unit used in the duration 'fracDur';
- 'value(fracDur)' is the fractional value used with the duration 'fracDur';
- 'duration(date1, date2)' is a function to calculate the duration between two dates or times (see [D.4.3](#) and [D.4.4](#)).

Given that 'date + unit(fracDur)' can be calculated, this method always produces a consistent value.

EXAMPLE 2 Given the expression: '2018-01-23 + P0.5M':

- It is rephrased as 'duration(2018-01-23 + P1M, 2018-01-23) × 0.5 + 2018-01-23'.
- It is reduced to 'duration(2018-02-23, 2018-01-23)'.
- Then 'P31D × 0.5 + 2018-01-23'.
- Then 'P15.5D + 2018-01-23'.
- Since 'P15.5D' is an exact duration (duration of calendar day is known), 'P15.5D + 2018-01-23' is resolvable and gives the final result '2018-02-07T12:00:00'.

D.4.5 Involving speculative duration

As described in [D.2.3](#), there is no universal rule to calculate leap seconds in advance, since the decision to insert a leap second is driven by a number of dynamic factors and only known when announced by the BIPM.

The rules specified in [D.2.3](#) are used to evaluate a date time formula that involves leap seconds.

EXAMPLE 1 This example applies identically with or without a leap second:

- Given the expression: '2018-12-31T23:59:59 + PT1M';
- Apply the durational unit clock minute directly to the representation: '2018-12-31T23:60:59';
- Since '59' is the last clock minute, treat the overflow with carry-over: '2018-12-31T24:00:59';
- Since '23' is the last clock hour, treat the overflow with carry-over: '2018-12-32T00:00:59';
- Since '31' is the last calendar day of year, treat the overflow with carry-over: '2018-13-01T00:00:59';
- Since '13' is the last calendar month, treat the overflow with carry-over: '2019-01-01T00:00:59';

- The resulting expression is '2019-01-01T00:00:59'.

EXAMPLE 2 Given the expression containing a leap second: '2016-12-31T23:59:60 + PT1M':

- Apply the durational unit clock minute directly to the representation: '2016-12-31T23:60:60';
- Since '59' is the last clock minute, treat the overflow with carry-over: '2016-12-31T24:00:60';
- Since '23' is the last clock hour, treat the overflow with carry-over: '2016-12-32T00:00:60';
- Since '31' is the last calendar day of year, treat the overflow with carry-over: '2016-13-01T00:00:60';
- Since '13' is the last calendar month, treat the overflow with carry-over: '2017-01-01T00:00:60';
- The clock second '60' is out of bounds, it is truncated to the maximum value: '2017-01-01T00:00:59';
- The resulting expression is '2017-01-01T00:00:59'.

EXAMPLE 3 Given the expression containing a leap second: '2016-12-31T23:59:59 + PT1S':

- Apply the durational unit clock second directly to the representation: '2016-12-31T23:59:60';
- The resulting expression is '2016-12-31T23:59:60'.

Bibliography

- [1] [ISO 80000-1](#)¹⁾, *Quantities and units — Part 1: General*
- [2] [ISO 80000-3](#)²⁾, *Quantities and units — Part 3: Space and time*
- [3] IETF RFC 5545, *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*
- [4] Library of Congress EDTF, The Library of Congress. *Extended Date/Time Format 1.0*. 2018-10-04. Available from: <http://www.loc.gov/standards/datetime/edtf.html> [viewed 2018-10-05]
- [5] CalConnect CC 18011, The Calendaring and Scheduling Consortium. CalConnect Standard: *Explicit date and time representation*. 2018-10-04
- [6] CalConnect CC 18012, The Calendaring and Scheduling Consortium. CalConnect Standard: *Generalized recurrence syntax*. 2018-10-04

1) Under preparation. Stage at the time of publication: ISO/DIS 80000-1:2019.

2) Under preparation. Stage at the time of publication: ISO/FDIS 80000-3:2019.

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards-based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Copyright in BSI publications

All the content in BSI publications, including British Standards, is the property of and copyrighted by BSI or some person or entity that owns copyright in the information used (such as the international standardization bodies) and has formally licensed such information to BSI for commercial publication and use.

Save for the provisions below, you may not transfer, share or disseminate any portion of the standard to any other person. You may not adapt, distribute, commercially exploit, or publicly display the standard or any portion thereof in any manner whatsoever without BSI's prior written consent.

Storing and using standards

Standards purchased in soft copy format:

- A British Standard purchased in soft copy format is licensed to a sole named user for personal or internal company use only.
- The standard may be stored on more than 1 device provided that it is accessible by the sole named user only and that only 1 copy is accessed at any one time.
- A single paper copy may be printed for personal or internal company use only.
- Standards purchased in hard copy format:
- A British Standard purchased in hard copy format is for personal or internal company use only.
- It may not be further reproduced – in any format – to create an additional copy. This includes scanning of the document.

If you need more than 1 copy of the document, or if you wish to share the document on an internal network, you can save money by choosing a subscription product (see 'Subscriptions').

Reproducing extracts

For permission to reproduce content from BSI publications contact the BSI Copyright & Licensing team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email subscriptions@bsigroup.com.

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Useful Contacts

Customer Services

Tel: +44 345 086 9001

Email (orders): orders@bsigroup.com

Email (enquiries): cservices@bsigroup.com

Subscriptions

Tel: +44 345 086 9001

Email: subscriptions@bsigroup.com

Knowledge Centre

Tel: +44 20 8996 7004

Email: knowledgecentre@bsigroup.com

Copyright & Licensing

Tel: +44 20 8996 7070

Email: copyright@bsigroup.com

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK